

References

- [1] L. Chen and et al. Data dependency in transformer-based object detection for medical imaging. *IEEE Transactions on Medical Imaging*, 42(5):1380–1390, 2023.
- [2] Philippa D Darbre. Recorded quadrant incidence of female breast cancer in great britain suggests a disproportionate increase in the upper outer quadrant of the breast. *Anticancer research*, 25(3C):2543–2550, 2005.
- [3] N. Dhungel, G. Carneiro, and A. P. Bradley. Deep learning with data augmentation for improved mammography classification. *IEEE Transactions on Medical Imaging*, 36(7):1699–1709, 2017.
- [4] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021.
- [5] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021.
- [6] Glenn Jocher and et al. Yolov8, 2023.
- [7] D. Lee and et al. Contrast sensitivity of transformer-based models in digital mammography versus screen-film mammography. *Radiology Artificial Intelligence*, 6(1):e230045, 2024.
- [8] L. Looney and et al. Breast cancer diagnosis in mammograms using deep learning and data augmentation. *Journal of Digital Imaging*, 32(3):543–552, 2019.
- [9] E. Denise Pisano, S. Zong, B. M. Hemminger, M. DeLuca, R. E. Johnston, Karl Muller, and et al. Contrast limited adaptive histogram equalization image processing to improve the detection of simulated spiculations in dense mammograms. *Journal of Digital Imaging*, 11(4):193–200, 1998.
- [10] Joseph Redmon and Ali Farhadi. Yolo9000: Better, faster, stronger. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7263–7272, 2017.
- [11] Villa Serena. Densità mammografica: che cosa vuol dire seno denso, 2022. GVM Care & Research, consultato il 20 luglio 2025.
- [12] W. Wang and et al. Rt-detr: Real-time transformer for object detection. *arXiv preprint arXiv:2304.08071*, 2023.
- [13] W. Wang and et al. Rt-detr: Real-time transformer for object detection. *arXiv preprint arXiv:2304.08071*, 2023.
- [14] Ryan W Woods, Gale S Sisney, Lonie R Salkowski, Kazuhiko Shinki, Yunzhi Lin, and Elizabeth S Burnside. The mammographic density of a mass is a significant predictor of breast cancer. *Radiology*, 258(2):417–425, 2011.
- [15] X. Zhu and et al. Deformable detr: Deformable transformers for end-to-end object detection. 2021.

STUDIO MAMMOGRAFIE

Progetto Fondamenti di Data Science e Machine Learning

Autori

Giuseppe Pio Sorrentino - 0522501965 - g.sorrentino101@studenti.unisa.it
Michele Pesce - 0522501963 - m.pesce13@studenti.unisa.it
Vito Piegari - 0522502037 - v.piegari2@studenti.unisa.it

Indice

1 Il Dataset MIAS (Mammographic Image Analysis Society)	5
1.1 Caratteristiche Principali	5
1.2 Esplorazione del dataset	6
1.3 Analisi di correlazione tra variabili	8
1.4 Determinazione di nuove feature	8
2 Predizione Testuale	11
2.1 Feature Extraction	13
2.1.1 Feature del dataset	13
2.1.2 Feature estratte	14
2.1.3 Correlazione - Apriori	15
2.2 Addestramento	16
2.2.1 Regressione Logistica	17
2.2.2 Random Forest	17
2.2.3 Support Vector Classifier (SVC)	18
2.2.4 VotingClassifier	19
2.3 Analisi dei risultati	20
3 Predizione sulle immagini	21
3.1 Pre-processing delle immagini	21
3.2 Data Augmentation	22
3.2.1 Tecniche di augmentation utilizzate	22
3.3 Modello di predizione	23
3.4 Risultati del modello	23
3.5 Considerazioni sui risultati	24
4 Modello multimodale	25
4.1 Primo test	25
4.2 Secondo Test	26
5 Object Detection	27
5.1 Preprocessing	27
5.1.1 Rimozione e pulizia delle feature testuali	27
5.1.2 Conversione delle feature in formato YOLO	28
5.1.3 Bilanciamento del dataset	29
5.1.4 Preprocessing e Pulizia delle Mammografie	30
5.2 Scelta dell'Architettura	41
5.3 Pipeline di Preparazione dei Dati per YOLOv8n-p2	45
5.3.1 Creazione della Struttura Dati YOLO e Trasformazione delle Annotazioni	45
5.3.2 Processamento Ulteriore delle Immagini	46
5.3.3 Aumentazione dei Dati: Espansione e Robustezza del Dataset	47
5.3.4 Suddivisione del Dataset in Set di Addestramento e Test	49
5.3.5 Configurazione del Dataset e Strategia di Addestramento del Modello YOLOv8n-p2	50

5.4	Valutazione performance del Modello	53
5.4.1	Metriche di Valutazione Dettagliate	53
5.4.2	Visualizzazioni dei Risultati di Training	56
5.4.3	Valutazione sul Test Set	57
5.5	Valutazione dell'Efficienza del Modello e Pipeline di Inferenza	57
5.5.1	Tempi di Esecuzione della Pipeline di Inferenza	59

1 Il Dataset MIAS (Mammographic Image Analysis Society)

Il dataset MIAS (Mammographic Image Analysis Society) è una risorsa ampiamente utilizzata nel campo dell'analisi delle immagini mediche, in particolare per la ricerca e lo sviluppo di algoritmi per la rilevazione e la classificazione delle anomalie nelle mammografie. Fornisce un insieme standardizzato di immagini e annotazioni, rendendolo prezioso per la valutazione di tecniche di elaborazione delle immagini e di intelligenza artificiale.

1.1 Caratteristiche Principali

- **Formato delle Immagini:** Le immagini sono tutte ridotte a una risoluzione uniforme di 1024×1024 pixel. Ogni pixel ha una dimensione fisica di 200 micron (equivalenti a 0,2 mm). Le immagini sono in scala di grigi (bianco e nero), una rappresentazione standard per le mammografie diagnostiche.
- **Annotazioni (Ground Truth):** Ogni mammografia nel dataset è corredata di annotazioni dettagliate che ne descrivono il contenuto clinico. Per ciascuna immagine, è indicato se contiene un'anomalia o se è considerata normale. In caso di anomalia, vengono fornite le seguenti informazioni:
 - Coordinate (x, y) del centro dell'anomalia.
 - La *classe* specifica dell'anomalia, che include 1:
 - * **CALC:** Calcificazioni.
 - * **CIRC:** Masse ben definite (circumscribed masses).
 - * **SPIC:** Masse spiculate.
 - * **MISC:** Masse mal definite (miscellaneous lesions).
 - * **ARCH:** Distorsioni architettoniche.
 - * **ASYM:** Asimmetrie.
 - * **NORM:** Nessuna anomalia evidente (normale).
 - Un'etichetta di *severità* che indica la natura dell'anomalia:
 - * **B:** Benigna.
 - * **M:** Maligna.
 - Un *raggio approssimativo* in pixel che definisce l'estensione dell'area della lesione.
- **Tessuto di Fondo:** Per ogni mammografia, è specificata la densità del tessuto di fondo mammario, un fattore importante per l'interpretazione clinica e l'analisi automatica:
 - **F (Fatty):** Tessuto prevalentemente adiposo (grasso).
 - **G (Fatty-glandular):** Tessuto misto adiposo e ghiandolare.
 - **D (Dense-glandular):** Tessuto prevalentemente denso e ghiandolare.
- **Numero di Immagini:** Il dataset comprende circa 322 immagini mammografiche. Queste immagini provengono da 161 pazienti, con le mammografie del seno sinistro e destro tipicamente fornite in coppia.

Il dataset MIAS è quindi uno strumento prezioso per la ricerca nel campo della diagnostica assistita da computer (CAD) per le mammografie, consentendo lo sviluppo e la validazione di metodi per la rilevazione precoce del cancro al seno.

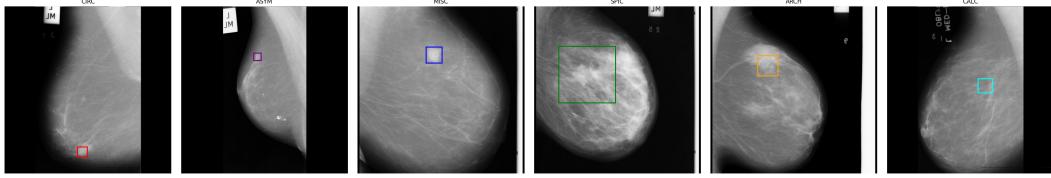


Figure 1: Anomalie viste dalle radiografie

1.2 Esplorazione del dataset

Seguono una serie di analisi sui dati presenti. Innanzi tutto osserviamo la distribuzione delle lesioni (benigne,maligne), come possiamo notare da 2 il dataset risulta leggermente sbilanciato verso i casi benigni Segue un confronto tra il tipo di anomalia e le occorrenze

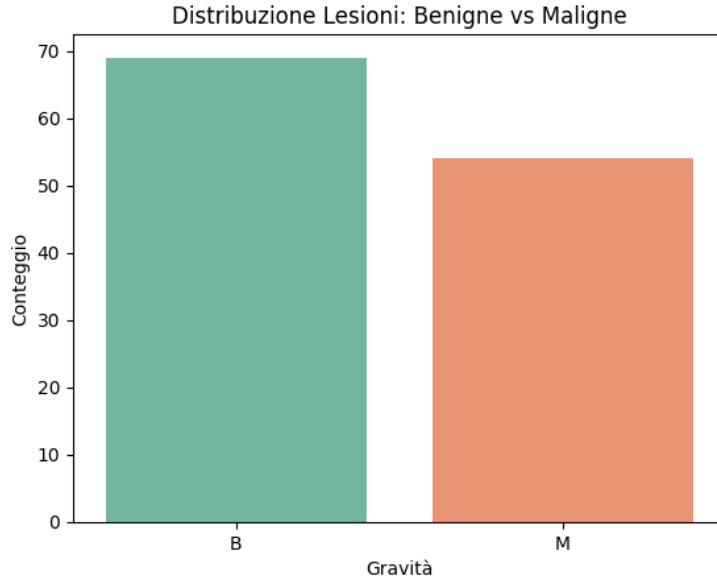


Figure 2: Anomalie confronto

di lesioni per ciascuna di esse. Come si può osservare dalla Figura 3, le classi **CALC** e **CIRC** sono quelle che, più di tutte, risultano associate alla presenza di tumore.

Una volta individuate le anomalie più frequenti, abbiamo analizzato la loro distribuzione in relazione al tipo di tessuto. Dall'analisi riportata in Figura 4, emerge che l'anomalia di tipo **CALC** si presenta circa nella metà dei casi nel tessuto di tipo D, mentre l'anomalia di tipo **CIRC** si manifesta, nella maggior parte dei casi, in tessuti di tipo F. 4

Un'ulteriore analisi è stata condotta sulle immagini, focalizzandosi sulla posizione spaziale delle lesioni. In particolare, è stata generata una *heatmap* che mostra la distribuzione dei centri (X, Y) delle anomalie individuate. Come illustrato in Figura 5, tale rappresentazione

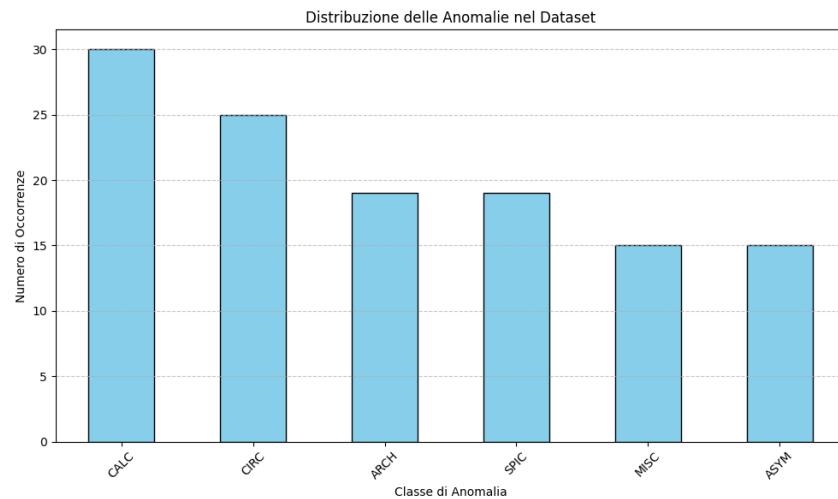


Figure 3: confronto tra tessuti

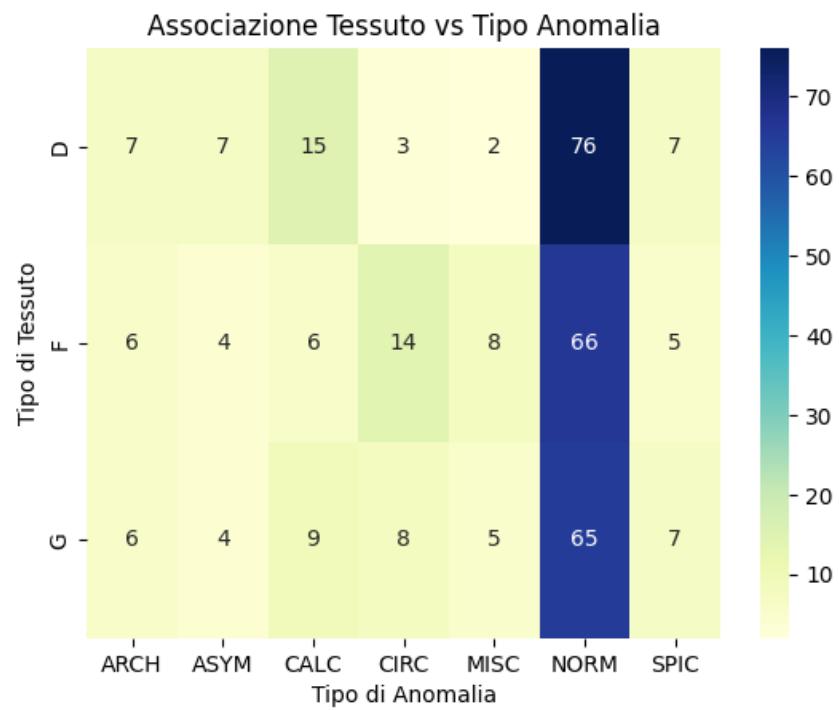


Figure 4: confronto tessuto - anomalie

permette di evidenziare le aree dell'immagine mammografica in cui si concentrano maggiormente le lesioni. Questa informazione può rivelarsi utile per eventuali strategie di pre-elaborazione o attenzione localizzata nei modelli di classificazione.

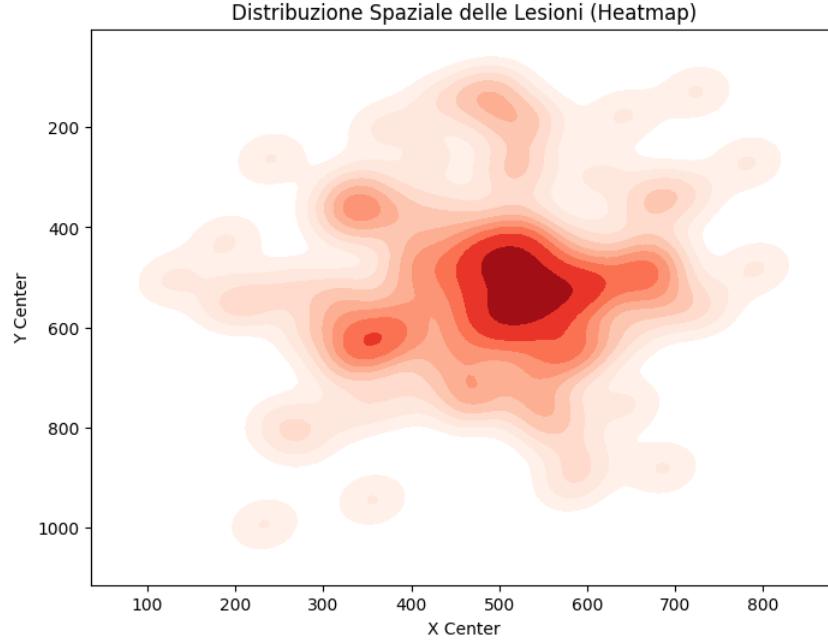


Figure 5: Distribuzione Spaziale centro anomalie

1.3 Analisi di correlazione tra variabili

Per comprendere meglio le relazioni tra le variabili del dataset, è stata condotta un'analisi di correlazione. Dopo un'adeguata fase di pulizia e pre-processing, sono stati selezionati i principali attributi numerici (coordinate del centro X , Y , raggio della lesione) e trasformate in variabili binarie (One-Hot Encoding) le variabili categoriali come *Severity*, *Breast Tissue* e *Anomaly Type*.

Sono state calcolate due matrici di correlazione:

- **Pearson**, per valutare relazioni lineari tra le variabili.
- **Spearman**, per individuare relazioni monotone anche non lineari.

Le relative heatmap sono riportate in Figura 6 e Figura 7, evidenziando eventuali pattern significativi tra tipo di tessuto, severità e localizzazione spaziale delle lesioni.

1.4 Determinazione di nuove feature

In questa fase ci siamo chiesti:

1. [RQ1]: Dal dataset MIAS è possibile determinare nuove features come *età* delle donne?

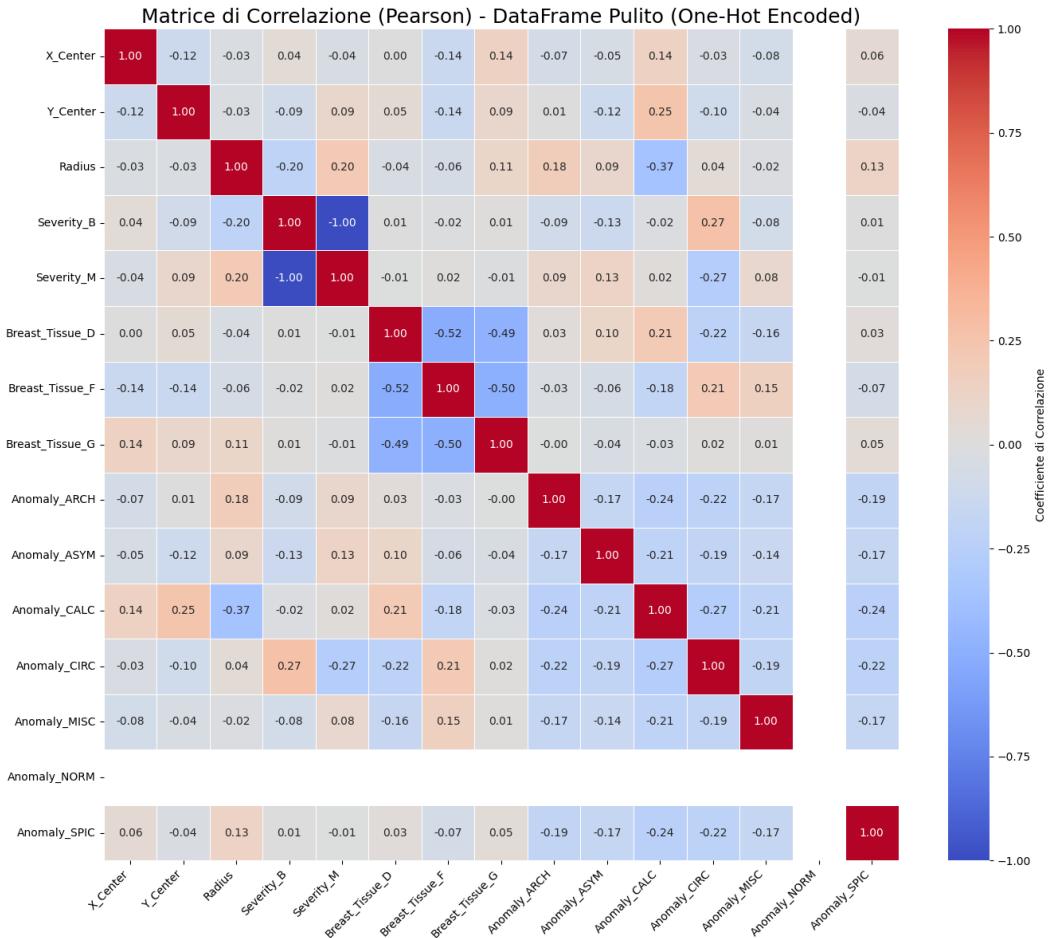


Figure 6: Matrice di correlazione (Pearson) tra le variabili numeriche e binarie del dataset

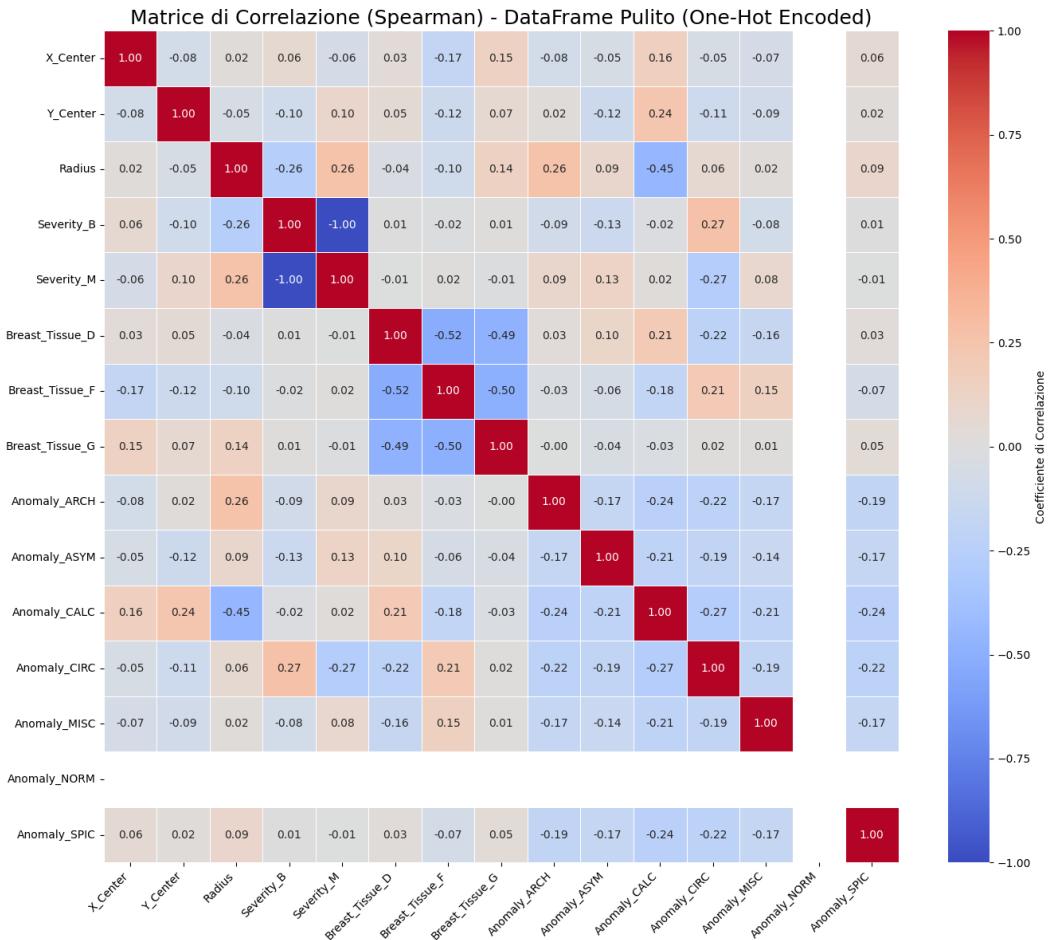


Figure 7: Matrice di correlazione (Spearman) tra le variabili numeriche e binarie del dataset



Figure 8: Diversi tipi di tessuto del seno

Un seno si definisce denso quando la quantità di tessuto ghiandolare prevale sulla componente adiposa. In linea generale il seno ha una struttura ghiandolare più densa quando non si sono ancora raggiunti i 40 anni mentre diventa meno denso quando la donna esce dal periodo fertile, con la menopausa, perché la quantità di tessuto ghiandolare diminuisce proporzionalmente all'avanzare dell'età e anche se si assumono farmaci contro le recidive del tumore al seno [11].

Come spiegato in precedenza, il nostro dataset distingue tre tipologie di tessuti: Fatty, Fatty-glandular e Dense-glandular. Come si può osservare nell'immagine 8, esiste una notevole somiglianza tra il seno denso ghiandolare (D) e il seno grasso ghiandolare (G). Inoltre, la densità di un seno è riconoscibile dall'estesa area bianca, che nel caso del tessuto denso copre oltre l'80% della mammella, risultando leggermente più ampia rispetto a quella del seno grasso ghiandolare. Infine, nella mammografia del seno grasso (Fatty), si nota l'assenza quasi totale di aree bianche.

Per poter effettuare questa analisi abbiamo utilizzato il dataset *mammography-breast-cancer-detection* presente su Kaggle. Come feature presenta *l'età* delle donne sottoposte alla mammografia, quindi abbiamo sfruttato questa caratteristica per poter plottare un esempio di seno per ogni età registrata nei metadati del dataset 9.

Dall'immagine si osserva che, con l'avanzare dell'età, vi è una marcata tendenza alla riduzione delle aree bianche nel tessuto mammario. Tuttavia, esistono numerosi casi che rendono **difficile** determinare l'età basandosi esclusivamente sulla presenza di tessuto ghiandolare. Di conseguenza, anche nel tentativo di stimare un intervallo di età utilizzando unicamente questa caratteristica, i risultati risulterebbero per lo più imprecisi.

Concludiamo quindi affermando che esiste una correlazione tra la composizione del tessuto ghiandolare e l'età, ma che sarebbero necessarie ulteriori informazioni per poter determinare quest'ultima con maggiore accuratezza.

2 Predizione Testuale

In questa sezione viene mostrato il processo di analisi e di valutazione svolto sulla parte testuale del dataset. L'obiettivo è stato quello di predire la categoria del tumore (Benzino/Maligno) basandosi esclusivamente sulle informazioni all'interno del file "mammografie.csv"

Esempio di Seno per Ogni Età (da 26 a 89)

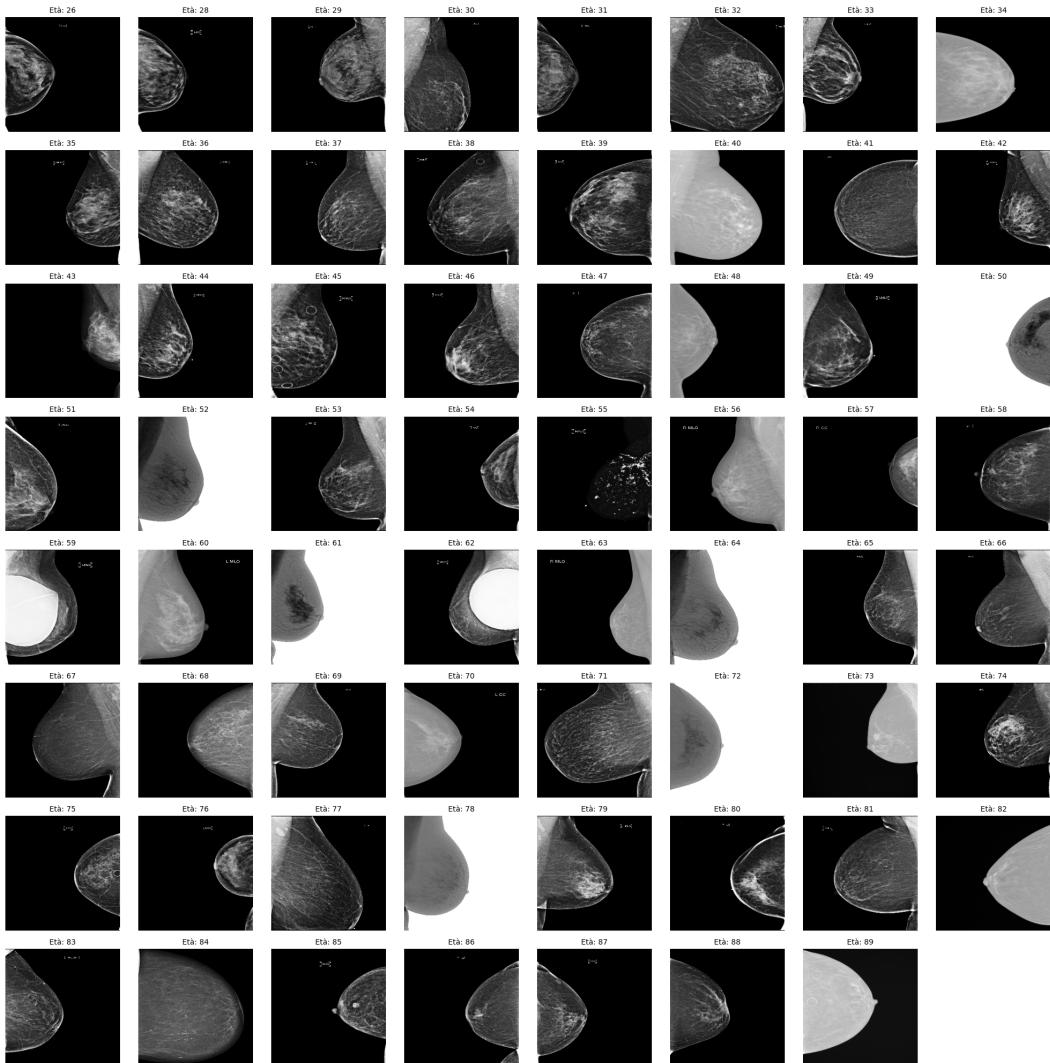


Figure 9: Mammografie plottate per età

2.1 Feature Extraction

L'estrazione delle feature rappresenta una fase cruciale nel processo di costruzione del modello predittivo, in quanto consente di trasformare i dati grezzi in variabili significative, interpretabili e utilizzabili dal classificatore. In questo studio, a partire dai metadati associati alle immagini mammografiche, sono state derivate diverse feature numeriche e categoriali, con l'obiettivo di catturare le principali dimensioni morfologiche, spaziali e contestuali della lesione. Le variabili selezionate alla fine sono un mix tra quelle già presenti all'interno del dataset e altre derivate da queste ultime.

2.1.1 Feature del dataset

Segue un elenco delle feature del dataset e un'analisi per ognuna di esse.

- **Tissue:** Tipologia del tessuto mammario presente nell'immagine. I valori comuni sono: F (Fatty), G (Glandular), D (Dense).
- **Class:** Forma morfologica della lesione osservata. Alcuni esempi comuni sono CIRC (circolare), SPIC (spiculated), MISC (miscellanea/irregolare), che rappresentano possibili indicatori di benignità o malignità. In particolare, nel nostro dataset, come riportato dal grafico 12, si osserva già una netta distinzione nei tumori di classe **CIRC** che tendono verso il benigno.

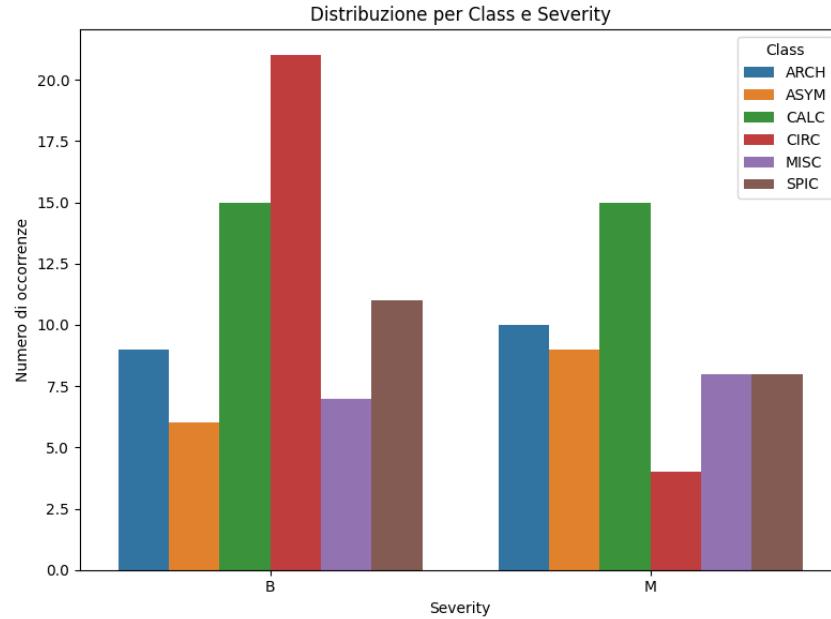


Figure 10: Distribuzione della Severity nel dataset

- **Severity:** Diagnosi clinica della lesione. B = Benigno, M = Maligno. È la variabile target per la classificazione. I tumori beigni risultano maggiori rispetto a quelli maligni.
- **X,Y:** Coordinate che indicano dove è presente il tumore.

- **Radius:** Raggio stimato dell'area della massa tumorale. Viene utilizzato per stimare la dimensione complessiva della lesione.

Dopo una serie di tentativi e valutazioni, la scelta è stata quella di escludere dall'analisi il raggio, nella sua misura che indica la lunghezza, e le coordinate X e Y essendo vaghe e poco impattanti ai fini di una predizione.

2.1.2 Feature estratte

Le informazioni analizzate nel precedente sottocapitolo, da sole, non hanno prodotto gli effetti sperati, nonostante i diversi approcci sperimentati con diversi classificatori quali il RandomForest, Regressione Lineare e SVC (Support vector classifier). Per rafforzare questi modelli predittivi la scelta è stata quella di introdurre delle nuove feature per avere maggiore precisione.

- **Area:** La dimensione della lesione è un noto indicatore radiologico associato alla probabilità di malignità. Tumori maligni tendono ad avere un'estensione maggiore rispetto ai benigni. Calcolare l'area a partire dal raggio consente di ottenere una misura continua e morfologicamente significativa. [14] Il Boxplot 11 ci dimostra infatti come, tendenzialmente, i tumori con un area maggiore risultano maligni a differenza dei benigni che presentano un area più piccola.

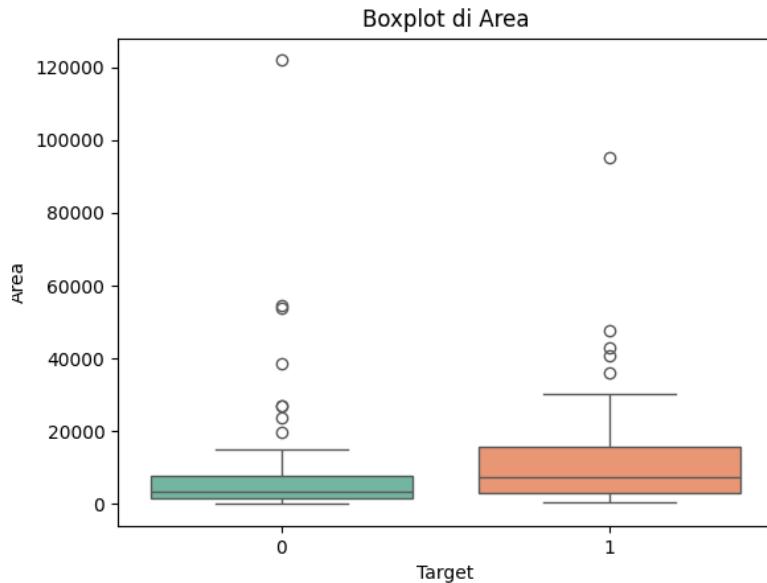


Figure 11: Boxplot area nel dataset

La tabella in questo caso ci dimostra come la differenza tra benigni e maligni sotto la media risulta abbastanza netta.

- **Quadranti:** In alternativa alle coordinate, la scelta è andata sull'utilizzo dei quadranti per classificare la zona di interesse. I quadranti nelle mammografie sono risultati di impatto nello studio delle zone in cui un tumore si presenta più frequentemente [2].

Table 1: Tabella di Contingenza Severità vs. Area della Massa

Severity	Sotto media	Sopra media
	B	M
B	41	28
M	23	31

Table 2: Distribuzione della Severità delle Masse per Quadrante e Lato

Quadrante	Lato	B	M
Q0	destro	8	9
	sinistro	8	1
Q1	destro	4	1
	sinistro	14	8
Q2	destro	14	12
	sinistro	1	8
Q3	destro	8	4
	sinistro	12	11

- **Distanza dal centro:** Rappresenta la distanza euclidea tra il centro della lesione (X, Y) e il centro geometrico dell'immagine mammografica (assunto a coordinate fisse (512, 512)).
- **RadiusBin:** trasformazione categoriale ordinata del raggio della massa tumorale (Radius), ottenuta tramite suddivisione in quintili dei valori osservati. I bin generati sono etichettati con valori da XS a XL, a indicare rispettivamente masse molto piccole fino a molto grandi. Questa discretizzazione consente al modello di cogliere eventuali effetti soglia o non lineari associati alla dimensione della lesione, semplificandone allo stesso tempo l'interpretazione clinica. L'analisi statistica tramite test chi-quadro ha confermato l'associazione tra le classi di grandezza (RadiusBin) e la diagnosi (benigno/maligno), con un p-value pari a 0.034, indicativo di una relazione significativa.

2.1.3 Correlazione - Apriori

Per approfondire le relazioni tra le feature estratte e quelle già esistenti, è stato impiegato l'algoritmo Apriori, con l'obiettivo di identificare pattern ricorrenti e regole di associazione significative tra variabili sia esplicite (ad esempio, classe e severità della lesione) che derivate (come il quadrante anatomico o la dimensione della lesione).

Le feature categoriali sono state codificate in forma transazionale e convertite in formato one-hot per consentire l'elaborazione. L'algoritmo Apriori è stato applicato con una soglia minima di supporto del 10%, e successivamente sono state estratte regole di associazione con un valore minimo di confidence pari al 60%. La tabella 3 ci mostra, come già deducibile dalle precedenti analisi, la possibilità molto alta di classificare tumori come benigni grazie alla qualità delle feature individuate. Possiamo notare infatti, come, tumori XS e di tessuto Circ tendano quasi sempre a rappresentare un caso benigno. Dalla tabella evince anche che il

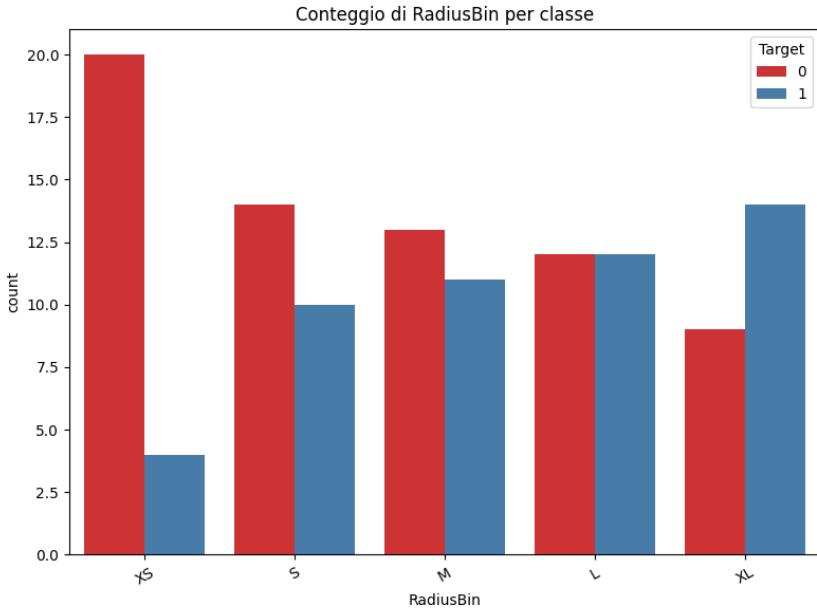


Figure 12: Distribuzione di B/M in base al RadiusBin

Table 3: Principali regole di associazione estratte con algoritmo Apriori (ordinate per *lift*)

Antecedente	Consequente	Supporto	Confidenza	Lift
Class=CALC, Severity=0	RadiusBin=XS	0.1008	0.8000	3.97
RadiusBin=XS, Severity=0	Class=CALC	0.1008	0.6000	2.64
Class=CALC, RadiusBin=XS	Severity=0	0.1008	0.8571	1.50
Class=CIRC	Severity=0	0.1681	0.8333	1.46
RadiusBin=XS	Severity=0	0.1681	0.8333	1.46
RadiusBin=XL	Severity=1	0.1176	0.6087	1.42
Quadrante=Q1	Severity=0	0.1513	0.6667	1.17
Quadrante=Q0	Severity=0	0.1345	0.6154	1.08
Quadrante=Q3	Severity=0	0.1597	0.6129	1.07

tessuto CALC, in combinazione con un Radius XS restituiscono con buona probabilità esito benigno, un tumore benigno di dimensione XS invece, nella quasi metà delle volte risulta appartenere alla classe CALC.. Come già analizzato in precedenza, l'algoritmo Apriori rafforza l'idea che lesioni più grandi implichino un tumore maligno.

2.2 Addestramento

Al fine di valutare l'efficacia predittiva delle feature estratte, è stato condotto un processo di addestramento supervisionato utilizzando diversi algoritmi di classificazione. In particolare, sono stati selezionati quattro modelli: Support Vector Classifier (SVC), Random Forest, Regressione Logistica e un classificatore ensemble basato su majority voting (VotingClassifier).

Ciascun modello è stato addestrato su un set di dati pre-processato e validato tramite una procedura di validazione incrociata a 10 fold, in modo da ottenere stime robuste delle performance generali. Per ogni classificatore, sono stati riportati indicatori standard come

accuracy, precision, recall, F1-score e matrice di confusione. L’obiettivo è confrontare le capacità discriminative dei modelli nel distinguere tra lesioni benigne e maligne a partire dalle variabili estratte.

Nel seguito, si presentano i risultati ottenuti per ciascun algoritmo, mettendone in luce le prestazioni sia sul test set che nella validazione incrociata.

2.2.1 Regressione Logistica

La regressione logistica è un modello lineare ampiamente utilizzato per compiti di classificazione binaria. Dal punto di vista matematico, essa modella la probabilità che un’osservazione appartenga a una determinata classe utilizzando una funzione logistica (sigmoide) applicata a una combinazione lineare delle variabili indipendenti.

Nel contesto del presente studio, il classificatore è stato addestrato su un dataset bilanciato mediante pipeline che includeva la standardizzazione delle feature numeriche e la codifica one-hot di quelle categoriali. Il modello ha raggiunto un’**accuratezza del 58.3%** sul test set, evidenziando un comportamento non ottimale nella distinzione tra le due classi. In particolare, la *recall* per la classe maligna è risultata pari al 70%, segno che il modello ha saputo identificare la maggior parte dei tumori maligni. Tuttavia, questa performance è stata ottenuta al prezzo di una maggiore quantità di falsi positivi, con una precisione del 50% per la classe benigna.

La matrice di confusione mostra infatti 7 casi benigni classificati erroneamente come maligni, indicando un certo sbilanciamento del modello a favore della classe maligna. Questa tendenza può derivare dalla natura lineare del classificatore, che potrebbe non essere in grado di catturare relazioni non lineari più complesse presenti tra le feature ingegnerizzate (es. Distanza dal centro o Area).

Dal punto di vista della validazione incrociata, il modello ha ottenuto un’accuratezza media di 0.580 con una deviazione standard di 0.084. Questo suggerisce una performance stabile ma non particolarmente elevata, coerente con i limiti intrinseci del modello su dataset a bassa dimensionalità e complessità.

	Predotto: Benigno	Predotto: Maligno
Reale: Benigno	7	7
Reale: Maligno	3	7

Table 4: Matrice di confusione - Regressione Logistica

2.2.2 Random Forest

La Random Forest è un algoritmo di apprendimento supervisionato basato su un insieme di alberi decisionali addestrati in modo casuale. La sua forza risiede nella combinazione di molteplici modelli deboli (decision tree) attraverso il metodo del *bagging* per ridurre l’overfitting tipico dei singoli alberi e migliorare la generalizzazione. Ogni albero della foresta è costruito su un sottoinsieme casuale del dataset e considera solo un sottoinsieme casuale delle feature a ogni nodo, favorendo la diversità interna.

Nel nostro studio, la Random Forest ha mostrato buone capacità discriminative, ottenendo un’**accuratezza del 66.7%** sul test set. In particolare, la *recall* della classe maligna

ha raggiunto l'80%, dimostrando un'efficacia nel rilevare correttamente i casi di tumore maligno. Tuttavia, si osserva una certa tendenza del modello a classificare erroneamente casi benigni come maligni.

Questo comportamento suggerisce che il modello sia orientato verso la massimizzazione della *recall* per la classe minoritaria (maligna), al prezzo di un incremento nei falsi positivi. Si tratta di un compromesso che può risultare accettabile in contesti clinici, dove il costo di un falso negativo è più elevato rispetto a quello di un falso positivo. La precisione per la classe benigna è infatti più alta rispetto alla regressione logistica, ma la capacità complessiva del modello resta intermedia.

La validazione incrociata a 10 fold ha restituito un'**accuracy media di 0.521** con una deviazione standard pari a 0.097, leggermente inferiore al risultato ottenuto sul test set. Questa discrepanza potrebbe indicare una leggera tendenza all'overfitting, nonostante la robustezza intrinseca del modello.

	Predetto: Benigno	Predetto: Maligno
Reale: Benigno	8	6
Reale: Maligno	2	8

Table 5: Matrice di confusione - Random Forest

2.2.3 Support Vector Classifier (SVC)

Il Support Vector Classifier (SVC) è un modello basato sulla teoria dei margini massimi, che mira a trovare l'iperpiano che separa le classi con la massima distanza possibile dai punti più vicini (support vectors). Quando i dati non sono linearmente separabili, l'algoritmo può proiettare gli esempi in uno spazio ad alta dimensione tramite kernel, rendendo possibile una separazione più efficace. L'SVC è noto per la sua capacità di gestire problemi complessi con buone proprietà di generalizzazione, anche in presenza di dati con rumore moderato.

Nel contesto di questo studio, l'SVC ha fornito le prestazioni migliori tra tutti i modelli analizzati, raggiungendo un'**accuratezza del 75%** sul test set. La matrice di confusione mostra una distribuzione equilibrata degli errori. La *recall* della classe maligna è risultata pari all'80%, e quella della classe benigna al 71%, con valori di *precision* superiori rispetto agli altri modelli. Questo dimostra una capacità notevole del modello di distinguere entrambe le classi, mantenendo un buon equilibrio tra sensibilità e specificità.

Una possibile spiegazione di tale performance risiede nella capacità dell'SVC di modellare relazioni non lineari tra le feature, sfruttando implicitamente il contributo di feature ingegnerizzate come la Distanza dal centro o le categorie RadiusBin e Quadrante. Inoltre, l'SVC tende a essere meno influenzato dalla distribuzione sbilanciata delle classi, grazie all'uso della funzione di margine che penalizza maggiormente gli errori sui support vectors.

La validazione incrociata a 10 fold ha restituito un'**accuracy media pari a 0.545** (± 0.138), un valore più basso rispetto al risultato ottenuto sul test set, ma coerente con l'elevata varianza tipica di modelli più sensibili alla composizione del training set. Ciononostante, l'SVC si conferma il modello con la miglior combinazione di accuratezza, bilanciamento tra le classi e capacità di generalizzazione. Il grafico 13 ci mostra l'utilità di ogni feature per la classificazione

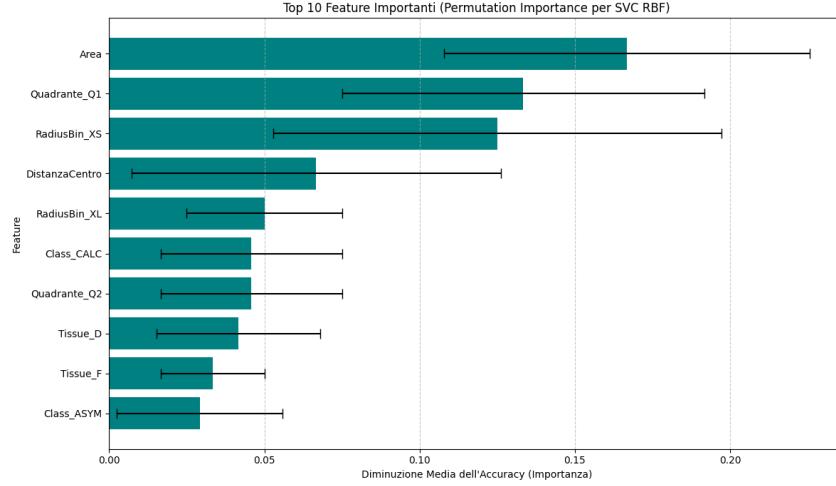


Figure 13: Feature più importanti nella SVC

	Predotto: Benigno	Predotto: Maligno
Reale: Benigno	10	4
Reale: Maligno	2	8

Table 6: Matrice di confusione - Support Vector Classifier

2.2.4 VotingClassifier

Il VotingClassifier è un modello ensemble che combina le predizioni di più classificatori base attraverso una strategia di voto, che può essere "hard" (classe più votata) o "soft" (somma delle probabilità). Nel presente lavoro è stato adottato un approccio di *soft voting*, che tiene conto delle probabilità di appartenenza alle classi fornite dai tre classificatori sottostanti: Support Vector Classifier, Random Forest e Regressione Logistica.

L'obiettivo del VotingClassifier è sfruttare la diversità tra modelli per aumentare la robustezza e la generalizzazione della predizione. Tuttavia, in questo caso, il classificatore ensemble ha mostrato le performance peggiori tra tutti i modelli testati. Sul test set ha ottenuto un'**accuracy del 54.2%**, con valori di precisione e richiamo modesti. In particolare, la *recall* per la classe maligna si è fermata al 50%, indicando che la metà dei tumori maligni sono stati erroneamente classificati come benigni.

La matrice di confusione riflette un'incapacità del modello di trovare un equilibrio efficace tra le due classi. La scarsa complementarità tra i modelli base potrebbe aver limitato il beneficio dell'ensemble: se gli errori commessi dai singoli classificatori sono simili, il modello aggregato rischia di amplificarli invece che correggerli.

La validazione incrociata ha confermato la bassa performance media, con un'**accuracy media pari a 0.571** e una deviazione standard di 0.087. Ciò indica anche una certa instabilità tra i diversi fold, segno che il modello non riesce a generalizzare efficacemente.

	Predetto: Benigno	Predetto: Maligno
Reale: Benigno	8	6
Reale: Maligno	5	5

Table 7: Matrice di confusione - VotingClassifier

2.3 Analisi dei risultati

In conclusione quindi la tabella 8 mostra una comparazione dei parametri forniti dai diversi classificatori.

Table 8: Confronto delle metriche di diversi modelli

Modello	Accuracy (Test)	Recall Maligno	Precision Benigno	CrossVal (Mean ±)
SVC	0.75	0.80	0.83	0.545 ± 0.138
Random Forest	0.67	0.80	0.80	0.521 ± 0.097
Regessione Logistica	0.58	0.70	0.70	0.580 ± 0.084
VotingClassifier	0.54	0.50	0.62	0.571 ± 0.087

L'analisi comparativa dei modelli evidenzia chiaramente la superiorità del Support Vector Classifier (SVC) nel contesto del presente dataset. Non solo ha raggiunto l'accuratezza più elevata (75%), ma ha anche mostrato un bilanciamento notevole tra le due classi, con valori di precisione e richiamo complessivamente migliori rispetto agli altri classificatori.

La Random Forest ha offerto buone prestazioni, specialmente nel riconoscimento dei casi maligni, ma con un incremento del tasso di falsi positivi che ne ha compromesso la precisione sulla classe benigna. La Regessione Logistica, pur semplice e interpretabile, ha mostrato limiti nell'adattarsi alla complessità del dataset, non riuscendo a superare il 60% di accuratezza. Il VotingClassifier, sebbene concettualmente promettente, ha fallito nell'integrare efficacemente le predizioni dei modelli di base, probabilmente a causa della loro bassa complementarità.

Nel complesso, l'SVC si conferma il miglior modello per questo compito specifico, grazie alla sua capacità di modellare relazioni non lineari e alla solidità delle sue prestazioni anche in presenza di feature miste (numeriche e categoriali).

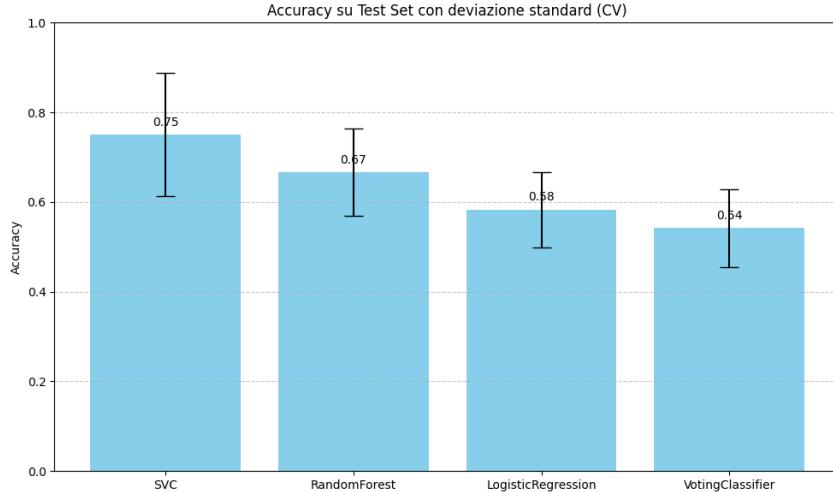


Figure 14: Confronto Accuracy dei modelli

3 Predizione sulle immagini

In questa sezione viene descritto il processo di previsione effettuato sulle immagini presenti nel dataset, con l’obiettivo di determinare la natura del tumore — se benigno o maligno. Le immagini, opportunamente preprocessate, sono state fornite in input a un modello di apprendimento addestrato per il riconoscimento di caratteristiche visive associate a ciascuna delle due classi.

Il sistema di classificazione sviluppato sfrutta tecniche di deep learning, in particolare una rete neurale convoluzionale (CNN), che si è dimostrata abbastanza efficace nel trattamento di dati visivi. Dopo la fase di addestramento, il modello è stato testato su un dataset di immagini di validation non viste durante l’apprendimento, al fine di valutarne la capacità predittiva.

3.1 Pre-processing delle immagini

Il pre-processing delle immagini rappresenta una fase fondamentale all’interno della pipeline di classificazione, in quanto permette di normalizzare e standardizzare i dati grezzi per l’input nel modello.

Ogni immagine viene sottoposta a una serie di trasformazioni atte a migliorarne la qualità e l’uniformità. La prima operazione consiste nel ritaglio dell’area di interesse, ovvero la regione corrispondente al tessuto mammario, a tale scopo, è stata implementata una funzione di cropping basata sulla distinzione automatica tra elementi di sfondo e elementi in primo piano, ottenuta tramite l’utilizzo di *cv2.THRESH_OTSU*.

Successivamente, i colori dell’immagine vengono invertiti (in modo che le masse siano più evidenti).

In seguito al cropping, per migliorare il contrasto e facilitare l’estrazione di pattern visivi, si applica la tecnica *CLAHE* (Contrast Limited Adaptive Histogram Equalization), utile in contesti medici per evidenziare dettagli nelle mammografie [9].

Questo processo assicura che tutte le immagini fornite al modello presentino caratteristiche coerenti, migliorando le prestazioni della rete e la qualità dell’apprendimento.

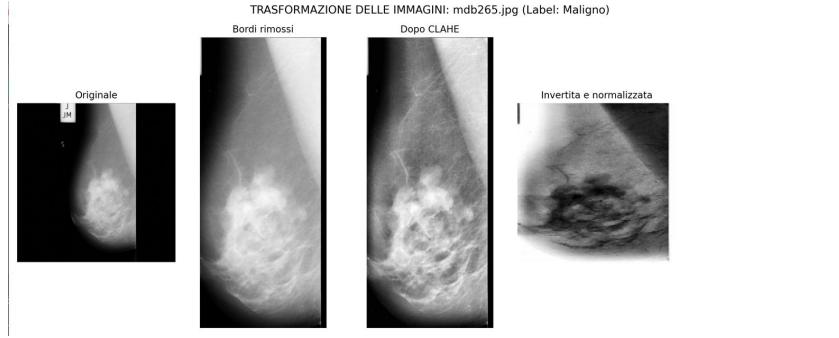


Figure 15: setps del preprocessing effettuato su ogni immagine

3.2 Data Augmentation

Per ridurre la possibilità di overfitting e migliorare la capacità di generalizzazione del modello, è stata implementata una procedura di data augmentation sulle immagini del dataset. Questa tecnica consiste nella generazione di più versioni modificate della stessa immagine, simulando variabilità realistiche che possono verificarsi nel mondo reale.

Per ogni immagine originale vengono generate **4** versioni aumentate (numero trovato dopo diversi test volti ad intetificare il massimo numero di copie da creare per evitare overfitting).

3.2.1 Tecniche di augmentation utilizzate

Dopo un'attenta analisi dello stato dell'arte per quanto riguarda l'utilizzo in ambito medico di tecniche di augmentation [3, 8], abbiamo deciso di utilizzare:

- **Rotazioni limitate:** rotazioni casuali entro un intervallo ristretto ($\pm 15^\circ$) per simulare variazioni fisiologiche nella posizione del paziente durante l'acquisizione. [8]
- **Flipping orizzontale:** utilizzata in modo condizionato per non invertire informazioni laterali (es. seno destro vs sinistro).
- **Zoom leggero:** ingrandimento casuale del contenuto dell'immagine mantenendo l'area visiva centrale.
- **Variazioni di contrasto:** variazioni casuali alla luminosità e al contrasto dell'immagine.

In una fase iniziale dell'esperimento erano state testate anche: Traslazioni casuali dell'immagine e Gaussian blur (una sfocatura leggera), tuttavia, queste due trasformazioni si sono rivelate controproducenti: nei risultati preliminari, hanno introdotto un rumore eccessivo che confondeva il modello, portando a un peggioramento delle metriche di classificazione.

Per questo motivo, tali trasformazioni sono state rimosse dalla pipeline finale, al fine di privilegiare aumentazioni conservative e coerenti con la natura clinica dei dati.

Durante la fase di validazione o test, invece, viene utilizzata solo la versione originale dell'immagine, senza augmentation, per garantire una valutazione obiettiva e riproducibile delle prestazioni del modello.

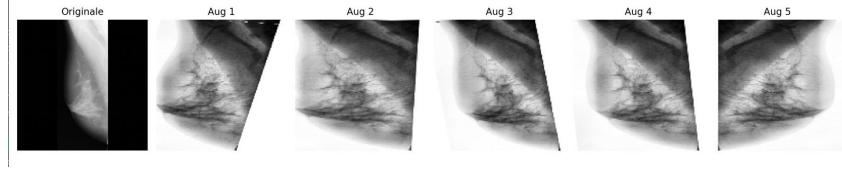


Figure 16: Esempio di augmentations per una immagine

3.3 Modello di predizione

È stato sviluppato un modello predittivo basato su una Convolutional Neural Network (CNN), progettato per analizzare immagini mammografiche in scala di grigi.

La rete è composta da tre blocchi convoluzionali, ciascuno con: convoluzione (kernel 3×3 , attivazione ReLU), batch normalization, max pooling (2×2) e dropout (25%). Dopo l'estrazione delle feature, segue uno strato fully connected con 64 neuroni (ReLU¹) e un dropout del 50%, seguito da un output sigmoid per la classificazione binaria.

3.4 Risultati del modello

Sono state sperimentate diverse strategie durante la costruzione del modello per migliorarne al massimo le prestazioni:

- **SMOTE:** l'utilizzo di tecniche di oversampling sintetico (SMOTE) non ha portato benefici, peggiorando le prestazioni complessive.
- **Oversampling:** è stata testata la duplicazione casuale delle immagini della classe minoritaria fino a un rapporto di 1 : 4 rispetto alla classe maggioritaria. Sebbene inizialmente sembrasse efficace, è stato osservato un aumento dell'*overfitting*, motivo per cui questa tecnica non è stata adottata nella versione finale del modello.
- **Bilanciamento dei pesi:** durante l'addestramento è stato applicato un bilanciamento dei pesi della loss function per compensare la distribuzione leggermente sbilanciata delle classi.
- **Preprocessing:** sono state valutate varie tecniche di normalizzazione e data augmentation per aumentare la robustezza del modello.
- **Early stopping:** è stato utilizzato con una `patience` pari a 5 epoch per evitare l'*overfitting*.

Il miglior modello è stato ottenuto dopo 5 epoch. Di seguito, le metriche di classificazione calcolate sulla validation set (36 immagini totali):

¹La funzione di attivazione ReLU è definita come: $ReLU(x) = \max(0, x)$.

Metriche utilizzate:

$$\begin{aligned}\text{Accuracy} &= \frac{TP + TN}{TP + TN + FP + FN} \\ \text{Precision} &= \frac{TP}{TP + FP} \\ \text{Recall} &= \frac{TP}{TP + FN} \\ \text{F1-score} &= 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}\end{aligned}$$

Metriche per il modello:

- **Classe Benigno** (21 esempi):

- Precision: 0.65
- Recall: 0.62
- F1-score: 0.63

- **Classe Maligno** (15 esempi):

- Precision: 0.50
- Recall: 0.53
- F1-score: 0.52

- **Accuracy complessiva:** 0.58

Matrice di confusione (validation set):

$$\begin{bmatrix} \text{Vero Benigno (TN)} & \text{Falso Maligno (FP)} \\ \text{Falso Benigno (FN)} & \text{Vero Maligno (TP)} \end{bmatrix} = \begin{bmatrix} 13 & 8 \\ 7 & 8 \end{bmatrix}$$

3.5 Considerazioni sui risultati

Il modello mostra prestazioni abbastanza scarse, con un'accuratezza del 58%. Si osserva una tendenza all'*overfitting*, nonostante l'adozione di tecniche di regolarizzazione.

Tuttavia, un aspetto relativamente positivo è che il modello tende a commettere più frequentemente errori a favore della classe **maligna**, il che, in ambito medico, è generalmente più accettabile rispetto a classificare come benigno un tumore che in realtà è maligno.

Per migliorare le prestazioni del modello e renderlo più adatto all'uso pratico, si è considerata l'integrazione di ulteriori informazioni durante l'addestramento attraverso l'utilizzo di modelli multimodali.

4 Modello multimodale

Alla luce dei risultati ottenuti utilizzando modelli basati esclusivamente su testo o immagini, si è ritenuto opportuno esplorare un approccio multimodale, che consenta di integrare entrambe le tipologie di dati. L’obiettivo è quello di sfruttare le sinergie tra le diverse modalità informative per migliorare le prestazioni complessive del sistema.

I modelli multimodali sono progettati per elaborare e combinare informazioni provenienti da più sorgenti, come testo, immagini, audio o video. Questa integrazione permette al modello di cogliere relazioni più complesse e profonde tra i dati, aumentando la capacità di generalizzazione rispetto ai modelli unimodali. In particolare, nel contesto di questo lavoro, si è adottato un modello in grado di coniugare rappresentazioni testuali e visive all’interno di un’unica architettura di apprendimento, al fine di valutare l’efficacia dell’approccio multimodale rispetto alle soluzioni precedenti.

4.1 Primo test

Nel primo tentativo di costruzione del modello, sono state selezionate come feature testuali le variabili **type** e **breast**, successivamente codificate tramite One-Hot Encoding all’interno di una matrice. Oltre all’One-Hot Encoding, non sono state effettuate ulteriori trasformazioni sulle feature testuali.

Per quanto riguarda le immagini, è stata eseguita una fase di preprocessing e augmentation come segue:

- **Preprocessing:** è stato effettuato un crop automatico per rimuovere loghi e sfondi non rilevanti intorno al seno, mantenendo un leggero padding per evitare tagli non desiderati. Il crop è stato basato sul contorno non nero più esteso presente nell’immagine. Successivamente, sono stati invertiti i colori e applicato il metodo CLAHE (Contrast Limited Adaptive Histogram Equalization) per migliorare il contrasto locale.
- **Augmentation:** per ogni immagine sono state generate 5 copie, ciascuna con una rotazione casuale compresa tra -10° e $+10^\circ$ e un flip orizzontale, entrambi applicati con probabilità del 50%.

Dopo la fase di preparazione dei dati, è stata costruita una rete neurale multimodale. La parte visiva è composta da una serie di layer convoluzionali, mentre la parte testuale è gestita da un layer denso seguito da un dropout. Le due rappresentazioni sono state successivamente concatenate e processate da ulteriori layer fully connected.

Il modello è stato addestrato per 30 epoche con *early stopping* e una *patience* di 10. Il miglior modello è stato ottenuto all’epoca 11, con i seguenti risultati sulle classi:

Classe	Precision	Recall	F1-score	Support
Benigno	0.89	0.76	0.82	21
Maligno	0.72	0.87	0.79	15
Accuracy			0.81	36

Table 9: Metriche di valutazione del modello al test (epoca 11).

4.2 Secondo Test

Il modello proposto, forte dei risultati del primo multimodale 4.1 e di quelli nell'SVC ??, abbiamo provato a combinare una rete neurale pre-addestrata con le feature testuali individuate precedentemente.

Preprocessing. I dati tabellari sono stati normalizzati con `StandardScaler` per le variabili numeriche, mentre le feature categoriali sono state trasformate mediante `OneHotEncoder`. Le immagini sono state preprocessate tramite una trasformazione personalizzata di ridimensionamento a 224×224 pixel, seguita da normalizzazione sui tre canali RGB. I dati sono stati organizzati in un oggetto `MultimodalDataset` contenente: l'immagine, il vettore delle feature e la classe. **L'Architettura del modello.** Il modello multimodale è composto da:

- Una **ResNet18** preaddestrata (con pesi ImageNet), privata dello strato fully connected finale. I suoi pesi sono stati congelati durante l'addestramento per preservare la rappresentazione visuale.
- Un **MLP** per il ramo tabellare, con due layer fully connected e funzione di attivazione ReLU.
- Un livello di **fusione multimodale** che concatena le embedding visive e tabellari, seguito da un classificatore finale a due classi.

Il dataset è stato suddiviso in 70% training e 30% validation. L'ottimizzazione è stata effettuata tramite **Adam**², che combina i vantaggi di due metodi precedenti: *AdaGrad*, che adatta il learning rate per ciascun parametro, e *RMSProp*, che utilizza una media esponenziale dei gradienti al quadrato, con learning rate $1 \cdot 10^{-3}$ e **CrossEntropyLoss**. È stata impiegata una **loss ponderata** per gestire la distribuzione sbilanciata delle classi, favorendo la classe Maligno.

Durante il training, il modello ha raggiunto una **accuratezza di validazione massima pari a 63.89%** all'*ottava epoca*, dove si è attivata l'early stopping, mentre andava chiaramente in *overfitting* nel training test. La metrica AUC finale è risultata pari a **0.7068**, indicando una discreta capacità discriminativa del classificatore.

Classification Report:

- **Benigni:** precisione 0.73, richiamo 0.44, F1-score 0.55
- **Maligni:** precisione 0.60, richiamo 0.83, F1-score 0.70

Questi valori evidenziano un comportamento sbilanciato: il modello mostra una maggiore sensibilità per la classe **Maligna** ($\text{recall} = 0.83$), il che è comunque desiderabile in ambito clinico, poiché permette di intercettare più casi critici. Tuttavia, questo avviene a scapito della precisione per i tumori benigni, che risultano frequentemente classificati in modo errato come maligni (falsi positivi).

Confusion Matrix:

$$\begin{bmatrix} 8 & 10 \\ 3 & 15 \end{bmatrix}$$

²(Adaptive Moment Estimation), un algoritmo stocastico per la discesa del gradiente ampiamente utilizzato per l'addestramento di reti neurali profonde.

La matrice di confusione conferma il comportamento osservato: dei 18 tumori benigni, solo 8 sono stati correttamente classificati, mentre 10 sono stati erroneamente etichettati come maligni. Al contrario, il modello ha identificato correttamente 15 tumori maligni su 18.

Il modello presenta una buona capacità di generalizzazione nella rilevazione dei tumori maligni, a discapito di un maggior tasso di falsi positivi. Il risultato è stato influenzato anche da feature ricavate dalla posizione del tumore, le quali hanno limitato la argumentation possibile sulle immagini a differenza del test 1 4.1.

5 Object Detection

In questa sezione affrontiamo un aspetto fondamentale del nostro progetto: la capacità di localizzare con precisione le anomalie presenti nelle immagini mammografiche.

Il dataset MIAS forniva informazioni essenziali per questo compito, in particolare le feature X-center, Y-center e Radius, che indicano rispettivamente le coordinate del centro e il raggio del cerchio che delinea la posizione e la dimensione dell'anomalia.

L'obiettivo di questa fase non era semplicemente classificare un'immagine come sana o anomala, ma spingersi oltre: sviluppare un modello di intelligenza artificiale in grado di individuare la posizione esatta delle anomalie, tracciando attorno ad esse dei bounding box e generando un'immagine di output che evidenzi visivamente tali regioni sospette.

5.1 Preprocessing

Per ottenere un modello altamente performante, è fondamentale che sia le immagini che le feature testuali siano adeguatamente preprocessate.

In questa fase descriveremo le tecniche di preprocessing applicate sia alle immagini mammografiche che ai dati testuali associati, illustrando brevemente i metodi utilizzati per pulire, normalizzare e rendere i dati più adatti all'addestramento del modello. Un preprocessing accurato è infatti cruciale per garantire che l'algoritmo apprenda dalle informazioni rilevanti, riducendo il rumore e migliorando l'accuratezza dei risultati.

5.1.1 Rimozione e pulizia delle feature testuali

Il preprocessing delle feature testuali è stato relativamente semplice ma essenziale per garantire la qualità dei dati in ingresso al modello. Il primo passo ha previsto la selezione delle colonne rilevanti, eliminando tutte le feature superflue. Abbiamo mantenuto solo le seguenti:

```
1 # Seleziona solo le colonne desiderate
2 columns_to_keep = ["Image_ID", "X_Center", "Y_Center", "Radius", "Anomaly_Type"]
3 df_filtered = df[columns_to_keep]
```

- **Image_ID**: Questa colonna è fondamentale per mantenere il collegamento tra le informazioni testuali e le immagini. Ogni valore rappresenta il nome di un'immagine (escludendo l'estensione) e si è rivelato utile per la generazione delle etichette in formato YOLO, in quanto consente di associare correttamente ogni riga del dataset alla sua immagine corrispondente.

- **X_Center, Y_Center, Radius:** Queste tre feature indicano la posizione e la dimensione delle anomalie, permettendo al modello di apprendere **dove** esse si trovano. Non tutte le immagini presentano questi valori (sono assenti nei casi senza anomalie), ma si sono dimostrati essenziali per convertire le coordinate nel formato normalizzato richiesto da YOLO (cioè **cx, cy, w, h**), come verrà descritto più avanti.
- **Anomaly_Type:** Questa colonna identifica la tipologia di anomalia presente. Originariamente distingueva tra varie classi di anomalie, ma ai fini del nostro modello era sufficiente distinguere tra casi **normali** e **anomali**. Per questo motivo, abbiamo trasformato tutti i valori diversi da **NORM** nel valore **ANORM**, semplificando così il problema in una classificazione binaria.

5.1.2 Conversione delle feature in formato YOLO

Per addestrare un modello di object detection in formato YOLO, è necessario convertire le informazioni sulle anomalie in una rappresentazione compatibile. In particolare, YOLO richiede che ogni oggetto/anomalia venga descritto da una bounding box normalizzata nel formato **[cx, cy, w, h]**, dove:

- **cx, cy** sono le coordinate normalizzate del centro della bounding box rispetto alla larghezza e altezza dell'immagine;
- **w, h** rappresentano larghezza e altezza, anch'esse normalizzate rispetto alle dimensioni totali dell'immagine.

Nel nostro caso, il dataset MIAS fornisce le feature **X_Center, Y_Center** e **Radius** che identificano il centro e la dimensione delle anomalie sotto forma di cerchio. Tali informazioni sono state convertite in bounding box quadrate che racchiudono l'anomalia, normalizzando infine le coordinate in modo che siano comprese nell'intervallo [0, 1].

Il cuore della conversione è riassunto nella seguente funzione Python:

```

1 def convert_to_normalized_cxcywh(row, img_width=1024, img_height=1024):
2     if pd.isna(row['X_Center']):
3         return None
4
5     x_center = row['X_Center']
6     y_center_img_coords = img_height - row['Y_Center']
7     radius = row['Radius']
8
9     xmin = max(0, x_center - radius)
10    ymin = max(0, y_center_img_coords - radius)
11    xmax = min(img_width, x_center + radius)
12    ymax = min(img_height, y_center_img_coords + radius)
13
14    width = xmax - xmin
15    height = ymax - ymin
16    cx = xmin + width / 2
17    cy = ymin + height / 2
18
19    return [cx / img_width, cy / img_height, width / img_width, height / img_height]

```

Questa funzione è stata applicata riga per riga a tutte le tuple contenenti anomalie, generando automaticamente le etichette nel formato YOLO da utilizzare per l'addestramento e la validazione del modello.

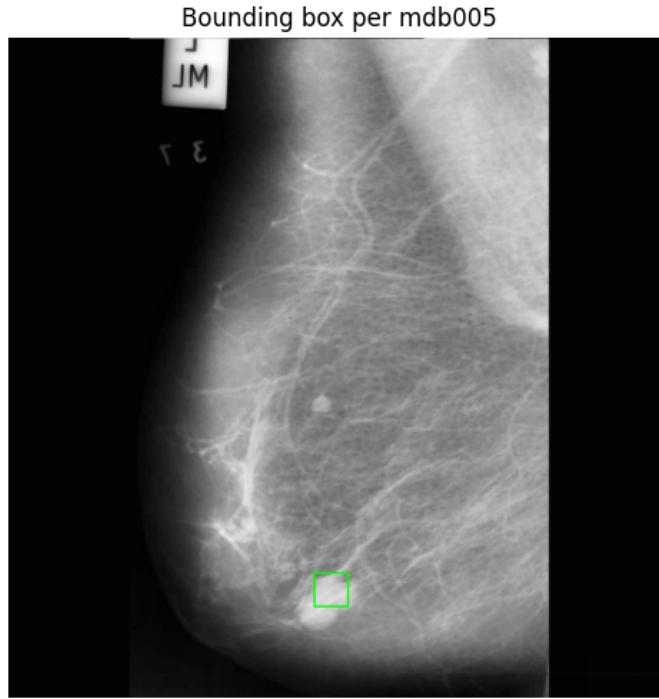


Figure 17: Esempio di immagine mammografica con bounding box generata a partire dalle coordinate fornite nel dataset MIAS.

5.1.3 Bilanciamento del dataset

Per garantire un addestramento equo del modello ed evitare bias verso la classe predominante, è stato necessario bilanciare il dataset. Nel dataset originale, infatti, era presente un'evidente disparità tra le immagini normali (**NORM**) e quelle contenenti anomalie (**ANORM**). Un dataset sbilanciato può compromettere le prestazioni del modello, portandolo a favorire la classe più rappresentata.

Il processo di bilanciamento ha previsto i seguenti passaggi principali:

- **Pulizia preliminare:** sono state rimosse le istanze con etichette errate o bounding box mancanti per le immagini ANORM.
- **Separazione delle classi:** il dataset è stato diviso in due sottoinsiemi, uno per ciascuna classe.
- **Undersampling:** per bilanciare il numero di esempi, la classe con più immagini è stata ridotta (tramite campionamento casuale) per egualare il numero della classe minoritaria.
- **Salvataggio dei dati bilanciati:** sia il file CSV aggiornato sia le immagini corrispondenti sono stati salvati in nuove directory dedicate.

Il risultato è un dataset bilanciato (**119 immagini per classe**) con un numero uniforme di esempi per ciascuna classe, come illustrato in Figura 18 – 19.

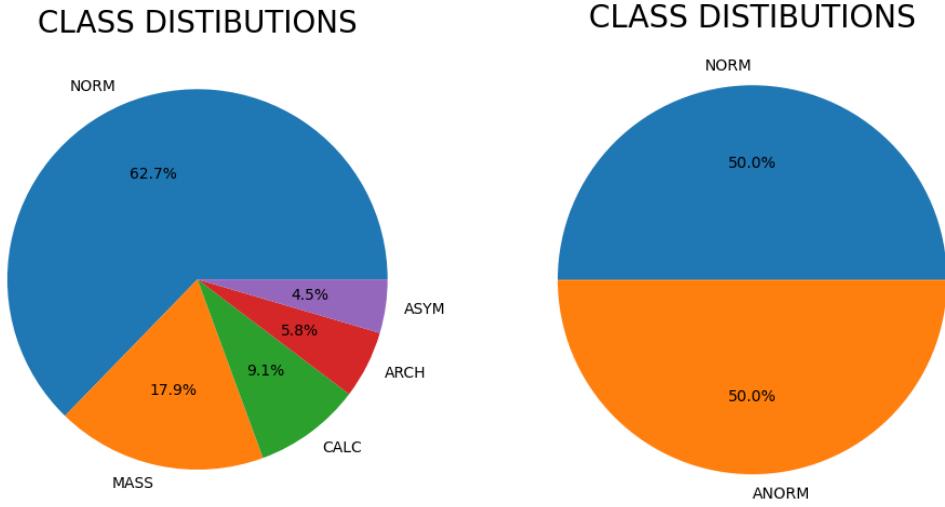


Figure 18: Distribuzione delle classi **prima** del bilanciamento.

Figure 19: Distribuzione delle classi **dopo** il bilanciamento.

5.1.4 Preprocessing e Pulizia delle Mammografie

Una fase fondamentale del nostro progetto è stata la pre-elaborazione delle mammografie per l'estrazione della **Regione di Interesse (ROI)** contenente il seno, priva di elementi di disturbo come etichette, bordi neri e muscolo pettorale. Il preprocessing è stato eseguito tramite una pipeline composta da tre macro-fasi: rimozione delle regioni non informative, miglioramento del contrasto e rimozione del muscolo pettorale.

1. Rimozione delle regioni non informative Il primo passo è consistito nel rimuovere bordi, sfondo e annotazioni presenti nelle immagini originali 20. Per farlo:

- **Normalizzazione dei pixel:** L'immagine mammografica viene inizialmente convertita in scala di grigi (se non lo è già) e rappresentata come array di numeri interi compresi tra 0 e 255, dove 0 corrisponde al nero e 255 al bianco. Per semplificare e rendere più stabili i calcoli numerici successivi, ogni valore di pixel viene normalizzato, ovvero diviso per 255, ottenendo così un'immagine dove i valori sono compresi tra 0 e 1. Questo processo permette, ad esempio, di confrontare direttamente i livelli di intensità tra immagini diverse, indipendentemente dalla loro esposizione.
- **Thresholding automatico (metodo di Otsu):** Una volta ottenuta l'immagine normalizzata, il passo successivo consiste nel separare le aree di interesse (cioè il seno) dallo sfondo tramite una tecnica chiamata *thresholding*, cioè sogliatura. In particolare, viene utilizzato il metodo di Otsu, che calcola automaticamente il valore di soglia ottimale analizzando l'istogramma dell'immagine. L'obiettivo è trovare un valore che divida i pixel in due classi (chiaro/scuro) in modo tale da massimizzare la differenza tra le due. Il risultato è un'immagine binaria: ogni pixel assume valore 0 (nero) se

sotto la soglia, o 255 (bianco) se sopra. In questo contesto, il seno risulta evidenziato in bianco, mentre lo sfondo viene rimosso (nero).

- **Pulizia tramite apertura morfologica:** L'immagine binaria ottenuta può contenere rumore: piccoli gruppi di pixel bianchi isolati che non fanno parte della struttura del seno. Per rimuovere queste imperfezioni, si applica una trasformazione chiamata *apertura morfologica*, una combinazione di erosione seguita da dilatazione. Questa operazione, usata frequentemente in elaborazione di immagini, serve a rimuovere piccole regioni bianche che non sono connesse a grandi strutture. In questo caso, si utilizza un elemento strutturante (detto *footprint*) a forma di disco con raggio 5. L'effetto finale è che le strutture piccole e isolate scompaiono, lasciando solo la forma più grande e coerente (il seno).
- **Auto-cropping orizzontale:** Dopo aver ottenuto una maschera binaria pulita, si passa alla fase di ritaglio automatico dell'immagine. In particolare, si analizza la prima riga della maschera binaria per determinare la posizione orizzontale dei primi e degli ultimi pixel bianchi. Questo permette di calcolare il punto da cui inizia e finisce la struttura del seno lungo l'asse orizzontale (da sinistra a destra). Viene quindi ritagliata l'immagine originale in corrispondenza di questi due estremi. Questo step è molto utile per rimuovere spazi vuoti inutili, ridurre le dimensioni dell'immagine e concentrare l'analisi solo sulla zona rilevante (il seno stesso).
- **Adattamento delle coordinate dei bounding box normalizzati:** Poiché le annotazioni dei bounding box sono espresse in formato normalizzato `cxcywh`, è necessario aggiornare la coordinata orizzontale *cx* nel caso in cui l'immagine venga ritagliata orizzontalmente su entrambi i lati.

Siano:

- W_{old} la larghezza dell'immagine originale;
- W_{new} la larghezza dell'immagine dopo il ritaglio;
- L il numero di pixel tagliati a sinistra;
- R il numero di pixel tagliati a destra.

La trasformazione corretta della coordinata normalizzata *cx* sarà:

$$cx' = \frac{cx \cdot W_{old} - L}{W_{new}} \quad (1)$$

La formula si interpreta così:

1. **Denormalizzazione:** $cx \cdot W_{old}$ converte la coordinata normalizzata in una coordinata assoluta;
2. **Shift:** $-L$ rimuove lo spostamento introdotto dal taglio a sinistra;
3. **Rinormalizzazione:** dividere per $W_{new} = W_{old} - L - R$ permette di adattare la coordinata al nuovo sistema di riferimento dell'immagine ritagliata.

In questo modo, anche dopo il doppio ritaglio laterale, i bounding box mantengono coerenza spaziale all'interno dell'immagine modificata.

Gli altri valori restano invariati (a differenza della larghezza del boundary box):

$$cy' = cy, \quad w' = \frac{w \cdot W_{old}}{W_{new}}, \quad h' = h \quad (2)$$

Questo adattamento garantisce che i bounding box continuino a rappresentare correttamente le regioni di interesse anche dopo il ritaglio, mantenendo la coerenza spaziale con l’immagine modificata.

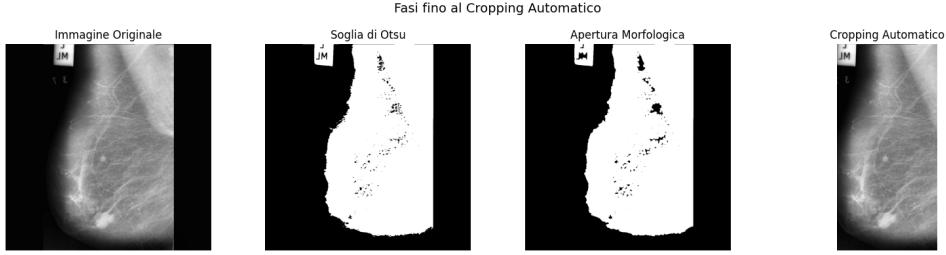


Figure 20: Fasi preliminari della pre-elaborazione dell’immagine mammografica. Da sinistra a destra: (1) immagine originale in scala di grigi, (2) sogliatura automatica mediante metodo di Otsu per separare tessuti dalla background, (3) apertura morfologica con elemento strutturante circolare per rimuovere piccoli oggetti e rumore, (4) ritaglio automatico (cropping) orizzontale per isolare la regione mammaria eliminando porzioni nere ai bordi.

2. Estrazione della Regione Mammaria (ROI) Dopo il ritaglio orizzontale e una nuova binarizzazione dell’immagine, si procede all’identificazione e all’estrazione della regione che rappresenta effettivamente il seno. Questo processo avviene tramite i seguenti passaggi fondamentali 21:

- **Nuova binarizzazione e apertura morfologica:** L’immagine ritagliata viene nuovamente sogliata con il metodo di Otsu per produrre una nuova maschera binaria. Su questa maschera si applica un’operazione di *apertura morfologica* utilizzando un elemento strutturante a forma di disco con raggio 5.
- **Etichettatura delle componenti connesse:** Dopo aver ottenuto una maschera binaria pulita, si procede con una tecnica chiamata *etichettatura delle componenti connesse* (*connected component labeling*). Questa operazione analizza l’immagine pixel per pixel e assegna un’etichetta numerica diversa a ciascuna “regione bianca” formata da pixel contigui (cioè adiacenti tra loro orizzontalmente, verticalmente o diagonalmente).

In altre parole, ogni gruppo di pixel bianchi che risulta connesso forma una **componente**, e il processo assegna un numero (es. 1, 2, 3, ...) a ciascuna di queste componenti. Ciascuna etichetta rappresenta un oggetto distinto all’interno dell’immagine: ad esempio, una può rappresentare la mammella, mentre altre possono essere piccoli artefatti, rumore o zone secondarie.

Questo passaggio è fondamentale perché consente di trattare separatamente ciascuna regione identificata e selezionare successivamente solo quella di interesse (tipicamente quella con area maggiore).

- **Selezione della regione con area massima:** Una volta che tutte le componenti connesse nella maschera binaria sono state etichettate, si procede con l'identificazione di quella che rappresenta effettivamente il seno. Per farlo, si analizzano tutte le componenti trovate e si calcola l'**area** di ciascuna, cioè il numero di pixel bianchi contenuti nella regione.

L'ipotesi alla base di questa scelta è che la regione più estesa (quella con più pixel bianchi) corrisponda al tessuto mammario vero e proprio, mentre le altre componenti (più piccole) sono verosimilmente rumore, artefatti o frammenti residui.

- **Creazione della maschera binaria finale:** Si crea una nuova maschera binaria contenente solo i pixel appartenenti alla regione selezionata, mentre tutte le altre regioni vengono annullate (poste a zero).
- **Applicazione della maschera all'immagine originale ritagliata:** La maschera binaria viene infine utilizzata per isolare la *Regione di Interesse (ROI)* all'interno dell'immagine ritagliata: tutti i pixel esterni alla regione selezionata vengono eliminati, mentre quelli interni vengono mantenuti invariati. Il risultato è un'immagine focalizzata unicamente sulla mammella, pronta per analisi successive.

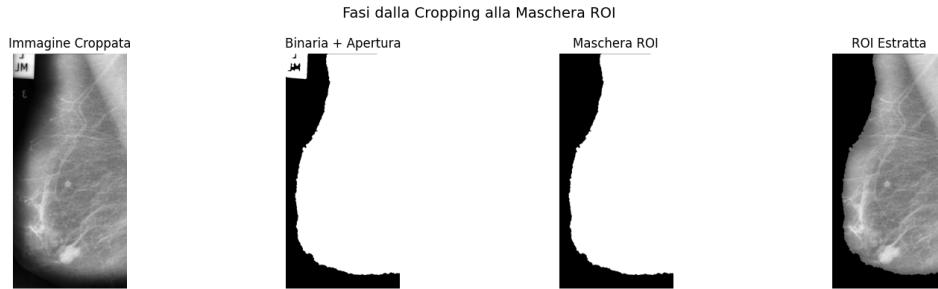


Figure 21: Sequenza di elaborazioni dalla fase di cropping alla generazione della maschera ROI. (1) Immagine ritagliata, (2) sogliatura e apertura morfologica per raffinare la regione mammaria, (3) maschera binaria contenente la regione connessa di area massima, (4) immagine originale filtrata con la maschera per ottenere la ROI estratta.

3. Miglioramento del contrasto Dopo l'estrazione della regione di interesse (ROI), è stato effettuato un miglioramento del contrasto dell'immagine, fondamentale per evidenziare dettagli strutturali e tessutali altrimenti poco visibili, specialmente in aree con intensità poco differenziata. Tale processo è stato suddiviso in due fasi distinte:

- **Filtro Mediano:** Per la riduzione del rumore impulsivo e di piccola scala è stato impiegato un filtro mediano, ovvero un filtro non lineare che opera su una finestra mobile locale (ad esempio di dimensione 3×3 o 5×5). Per ogni pixel, il filtro sostituisce il valore dell'intensità con la mediana dei valori presenti all'interno della finestra centrata su di esso.

A differenza di filtri lineari come il filtro medio, che tende a sfocare i contorni e le strutture, il filtro mediano preserva i bordi netti e i dettagli morfologici, rivelandosi particolarmente adatto in contesti medici. In mammografia, dove le microstrutture

e i margini delle lesioni possono avere importanza clinica, questo filtro consente una riduzione del rumore mantenendo la fedeltà diagnostica dell'immagine.

- **CLAHE (Contrast Limited Adaptive Histogram Equalization):** Dopo il filtraggio, è stato applicato il metodo CLAHE per un'equalizzazione del contrasto a livello locale. L'algoritmo CLAHE è una variante avanzata dell'equalizzazione dell'istogramma classica e si basa su tre principi fondamentali:

1. *Suddivisione in sotto-regioni (tiles)*: l'immagine viene divisa in piccole regioni locali di dimensione 8×8 (parametro `tileGridSize=(8, 8)`). In ciascun blocco viene calcolato l'istogramma e applicata l'equalizzazione in modo indipendente.
2. *Limitazione del contrasto (clip limit)*: per evitare l'eccessiva amplificazione del rumore in regioni omogenee, l'istogramma di ciascun blocco viene limitato in altezza. Nel codice, questo valore è impostato a `clipLimit=2.0`, che rappresenta un buon compromesso per immagini medicali. Le eccedenze oltre la soglia vengono redistribuite, mantenendo controllata la variazione locale.
3. *Interpolazione bilineare*: per evitare discontinuità tra le regioni adiacenti, i bordi dei blocchi vengono smussati mediante interpolazione bilineare, garantendo una transizione fluida e naturale del contrasto.

CLAHE è particolarmente adatto per immagini medicali con forti variazioni di densità tessutale, come le mammografie. Permette di migliorare il contrasto nelle aree a bassa intensità mantenendo stabili le zone già ben contrastate, migliorando così la leggibilità dei dettagli senza introdurre artefatti evidenti.

Tale combinazione di tecniche garantisce un'immagine più pulita e con contrasto ottimizzato, facilitando i successivi processi di elaborazione e interpretazione automatica.



4. Rimozione del muscolo pettorale È importante notare che, per standardizzare il processo e facilitare l'applicazione dell'algoritmo, tutte le immagini del dataset sono state inizialmente **rippate orizzontalmente**. Questo assicura che il muscolo pettorale, una struttura anatomica che può variare di posizione a seconda della proiezione della mammografia, si trovi sempre nell'angolo superiore destro dell'immagine, garantendo una posizione consistente per l'analisi.

Nelle mammografie, specialmente quelle acquisite in proiezione medio-laterale obliqua (MLO), il muscolo pettorale appare come una vasta regione di tessuto denso e luminoso

nell'angolo superiore destro dell'immagine. La sua rimozione è cruciale perché può oscurare potenziali anomalie o introdurre artefatti che potrebbero influenzare negativamente l'accuratezza degli algoritmi di rilevamento. Per questo scopo, viene adottata una strategia robusta che combina tecniche di segmentazione e analisi di regione 22:

- Il processo inizia applicando una **soglia multi-Otsu** all'immagine, la quale è stata precedentemente migliorata in contrasto. Questa tecnica avanzata di sogliatura non si limita a dividere l'immagine in due classi (come la soglia di Otsu standard), ma la segmenta in **tre classi distinte di intensità**: regioni scure, regioni a intensità media e regioni molto luminose. L'assunto fondamentale è che le aree più luminose dell'immagine sono quelle con maggiore densità, e quindi potenzialmente associate al muscolo pettorale o ad altri tessuti densi che si desidera isolare.
- Successivamente, la classe di intensità più elevata, identificata come la più luminosa tra le tre, viene estratta per formare una **maschera binaria preliminare**. Questa maschera è una rappresentazione approssimativa delle aree che si ritiene contengano il muscolo. Ogni pixel in questa maschera sarà "attivo" (rappresentato dal valore 1) se appartiene alla classe luminosa, e "inattivo" (0) altrimenti.
- Per affinare questa maschera preliminare e rimuovere il rumore, come piccole macchie o componenti isolate che non fanno parte del muscolo principale, viene applicata un'**apertura morfologica**. Questa operazione di elaborazione delle immagini, che utilizza un elemento strutturante a forma di disco, ha un duplice effetto: elimina piccoli oggetti e leviga i contorni delle regioni rimanenti. Il risultato è una maschera più pulita e coerente, che rappresenta in modo più fedele le regioni luminose significative.
- A questo punto, le diverse **componenti connesse** all'interno della maschera pulita vengono identificate e analizzate individualmente. Per ciascuna di queste regioni, vengono estratte proprietà come l'area e le coordinate del rettangolo di delimitazione (bounding box). Viene quindi selezionata la regione che soddisfa due criteri specifici per essere identificata come il muscolo pettorale: deve avere un'area sufficientemente grande (maggiore di 500 pixel) per escludere piccoli artefatti, e crucialmente, deve trovarsi in una **posizione ben definita nell'angolo superiore destro** dell'immagine. Questa posizione è geometricamente definita, ad esempio, richiedendo che la sua riga più alta sia entro il primo terzo superiore dell'immagine e la sua colonna più a sinistra sia oltre la metà della larghezza dell'immagine. Questo filtro posizionale è essenziale per isolare specificamente il muscolo pettorale da altre strutture dense.
- Se una regione che soddisfa questi criteri viene identificata come il muscolo, essa viene trasformata in una maschera binaria specifica per quella singola regione. L'immagine originale viene quindi modificata azzerando i valori di intensità di tutti i pixel corrispondenti a questa maschera: questa è la **rimozione semplice del muscolo**.
- Tuttavia, la segmentazione iniziale basata sulla soglia potrebbe non catturare l'intera estensione del muscolo, lasciando alcune zone chiare attigue o con intensità leggermente inferiori che sono ancora parte del muscolo. Per affrontare questa limitazione, viene applicata una **rimozione estesa**, una strategia più aggressiva e robusta.

- Per realizzare questa estensione, si inizia da un punto di partenza strategico: il **pixel nell'angolo in alto a destra** dell'immagine. Da lì, si avvia un processo di esplo-razione (simile a una ricerca in profondità o a un "flood fill"). Il sistema si espande verso i pixel vicini non ancora considerati, aggiungendoli a una nuova **maschera estesa**. Questa espansione continua finché non si incontra un pixel che fa parte della **maschera muscolare originale**. Questo punto di "incontro" funge da confine nat-urale del muscolo. La logica sottostante è che, partendo dall'angolo e "riempiendo" l'area, si includono tutte le regioni luminose e connesse che si trovano sopra e a destra del muscolo già rilevato, colmando eventuali **lacune o regioni adiacenti che erano state erroneamente escluse dalla sogliatura iniziale** ma che dovrebbero essere rimosse.
 - Il risultato di questa espansione è una **maschera estesa**, che comprende non solo il muscolo originariamente rilevato ma anche tutte le zone adiacenti che sono logicamente parte della regione da rimuovere.
 - Infine, l'immagine viene aggiornata nuovamente, azzerando tutti i pixel corrispondenti a questa nuova maschera estesa. Questo passaggio finale garantisce una rimozione del muscolo più accurata e completa, minimizzando la sua influenza sulle analisi successive.



Figure 22: Questa figura illustra il processo di rimozione del muscolo pettorale da una mammografia. Vengono mostrate le fasi: l'immagine originale, la rimozione semplice, la rimozione estesa e la maschera estesa finale utilizzata.

Codice Python del preprocessing su una sola immagine (la modifica dei boundary box è esclusa)

```
1 import cv2
2 import numpy as np
3 from skimage.morphology import disk, binary_opening
4 from skimage.measure import label, regionprops
5 from skimage.filters import threshold_multiotsu
6 import matplotlib.pyplot as plt
7 from skimage.filters import threshold_multiotsu
8
9 def preprocess_mammogram(image_path):
10     """
11         Applica una serie di tecniche di pre-elaborazione a una mammografia
12         e mostra quattro plot: fino al cropping automatico, fino alla maschera ROI,
13         miglioramento del contrasto, e rimozione del muscolo pettorale.
14     """
15     # Load the image
16     image = cv2.imread(image_path)
17
18     # Convert the image to grayscale
19     gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
20
21     # Apply binary opening to remove noise
22     binary = binary_opening(gray > 100)
23
24     # Label connected components
25     labels = label(binary)
26
27     # Find the largest component (the ROI)
28     roi_label = regionprops(labels)[0].label
29
30     # Extract the ROI
31     roi = labels == roi_label
32
33     # Apply thresholding to the ROI
34     threshold = threshold_multiotsu(roi)
35     binary_otsu = roi > threshold
36
37     # Invert the binary mask
38     binary_inverted = ~binary_otsu
39
40     # Show the results
41     plt.figure(figsize=(15, 5))
42     plt.subplot(1, 4, 1)
43     plt.imshow(image)
44     plt.title("Original Image")
45
46     plt.subplot(1, 4, 2)
47     plt.imshow(binary)
48     plt.title("Binary Opening")
49
50     plt.subplot(1, 4, 3)
51     plt.imshow(binary_inverted)
52     plt.title("Inverted Mask")
53
54     plt.subplot(1, 4, 4)
55     plt.imshow(binary_inverted * image)
56     plt.title("Cropped Image")
57
58     plt.show()
```

```

14      """
15      step_images_crop = []
16      step_titles_crop = []
17
18      step_images_roi = []
19      step_titles_roi = []
20
21      step_images_final = []
22      step_titles_final = []
23
24      step_images_muscle = []
25      step_titles_muscle = []
26
27      # 1. Caricamento dell'immagine in scala di grigi
28      img_original = cv2.imread(image_path, cv2.IMREAD_GRAYSCALE)
29      if img_original is None:
30          raise FileNotFoundError(f"Immagine non trovata o non leggibile: {image_path}")
31
32      step_images_crop.append(img_original)
33      step_titles_crop.append("Immagine Originale")
34
35      print("Passo 1: Rimozione Regioni Non Desiderate")
36
37      img_double = img_original.astype(np.float64) / 255.0
38      img_squared = (img_double * img_original).astype(np.uint8)
39
40      _, binary_initial = cv2.threshold(img_squared, 0, 255, cv2.THRESH_BINARY + cv2.THRESH_OTSU)
41      step_images_crop.append(binary_initial)
42      step_titles_crop.append("Soglia di Otsu")
43
44      footprint = disk(5)
45      binary_opened = binary_opening(binary_initial, footprint=footprint).astype(np.uint8) * 255
46      step_images_crop.append(binary_opened)
47      step_titles_crop.append("Apertura Morfologica")
48
49      first_row = binary_opened[0, :]
50      left_crop = np.argmax(first_row > 0)
51      right_crop = len(first_row) - np.argmax(first_row[::-1] > 0) - 1
52      img_cropped = img_original[:, left_crop:right_crop+1]
53      step_images_crop.append(img_cropped)
54      step_titles_crop.append("Cropping Automatico")
55
56      plt.figure(figsize=(16, 4))
57      for i, (img, title) in enumerate(zip(step_images_crop, step_titles_crop)):
58          plt.subplot(1, len(step_images_crop), i + 1)
59          plt.imshow(img, cmap='gray')
60          plt.title(title)
61          plt.axis("off")
62          plt.suptitle("Fasi fino al Cropping Automatico", fontsize=14)

```

```

63     plt.tight_layout()
64     plt.show()
65
66     print("Passo 1b: Estrazione della ROI")
67
68     step_images_roi.append(img_cropped)
69     step_titles_roi.append("Immagine Croppata")
70
71     _, binary_cropped = cv2.threshold(img_cropped, 0, 255, cv2.THRESH_BINARY + cv2.THRESH_OTSU)
72     binary_cropped_opened = binary_opening(binary_cropped, footprint=footprint).astype(np.uint8) * 255
73     step_images_roi.append(binary_cropped_opened)
74     step_titles_roi.append("Binaria + Apertura")
75
76     labeled_img = label(binary_cropped_opened)
77     regions = regionprops(labeled_img)
78
79     if not regions:
80         raise ValueError("Nessuna regione trovata nell'immagine binarizzata croppata..")
81
82     largest_region = max(regions, key=lambda x: x.area)
83     mask = np.zeros_like(binary_cropped_opened, dtype=np.uint8)
84     minr, minc, maxr, maxc = largest_region.bbox
85     mask[minr:maxr, minc:maxc] = (labeled_img[minr:maxr, minc:maxc] == largest_region.label).astype(np.uint8)
86     step_images_roi.append(mask)
87     step_titles_roi.append("Maschera ROI")
88
89     roi_extracted_img = cv2.bitwise_and(img_cropped, img_cropped, mask=mask)
90     step_images_roi.append(roi_extracted_img)
91     step_titles_roi.append("ROI Estratta")
92
93     plt.figure(figsize=(16, 4))
94     for i, (img, title) in enumerate(zip(step_images_roi, step_titles_roi)):
95         plt.subplot(1, len(step_images_roi), i + 1)
96         plt.imshow(img, cmap='gray')
97         plt.title(title)
98         plt.axis("off")
99     plt.suptitle("Fasi dalla Cropping alla Maschera ROI", fontsize=14)
100    plt.tight_layout()
101    plt.show()
102
103    print("Passo 2: Miglioramento del contrasto")
104
105    step_images_final.append(roi_extracted_img)
106    step_titles_final.append("ROI Estratta")
107
108    median_filtered = cv2.medianBlur(roi_extracted_img, 3)
109    step_images_final.append(median_filtered)
110    step_titles_final.append("Filtro Mediano 3x3")
111
```

```

112     clahe = cv2.createCLAHE(clipLimit=2.0, tileGridSize=(8, 8))
113     contrast_enhanced = clahe.apply(median_filtered)
114     step_images_final.append(contrast_enhanced)
115     step_titles_final.append("Contrasto Migliorato (CLAHE)")
116
117     plt.figure(figsize=(14, 4))
118     for i, (img, title) in enumerate(zip(step_images_final, step_titles_final)):
119         plt.subplot(1, len(step_images_final), i + 1)
120         plt.imshow(img, cmap='gray')
121         plt.title(title)
122         plt.axis("off")
123     plt.suptitle("Fasi dalla Maschera ROI al Miglioramento del Contrasto", fontsize=14)
124     plt.tight_layout()
125     plt.show()
126
127     # ----- RIMOZIONE MUSCOLO PETTORALE -----
128     print("Passo 3: Rimozione del muscolo pettorale (in alto a destra)")
129
130     thresholds = threshold_multotsu(contrast_enhanced, classes=3)
131     segmented = np.digitize(contrast_enhanced, bins=thresholds)
132
133     # Classe più luminosa = sospetta muscolo
134     bright_mask = (segmented == 2).astype(np.uint8) * 255
135
136     # Apertura morfologica per rimuovere piccole regioni
137     footprint = disk(5)
138     bright_mask_cleaned = binary_opening(bright_mask, footprint=footprint).astype(np.uint8) * 255
139
140     # Etichettatura
141     labeled = label(bright_mask_cleaned)
142     regions = regionprops(labeled)
143
144     muscle_removed = contrast_enhanced.copy()
145     muscle_removed_simple = contrast_enhanced.copy()
146     muscle_debug_mask = np.zeros_like(contrast_enhanced)
147     h, w = contrast_enhanced.shape
148     muscle_found = False
149
150     for region in regions:
151         minr, minc, maxr, maxc = region.bbox
152         in_upper_right = minr < h // 3 and minc > w // 2
153         if region.area > 500 and in_upper_right:
154             print(f" Regione muscolare trovata: area={region.area}, bbox={region.bbox}")
155             muscle_mask = (labeled == region.label).astype(np.uint8)
156
157             # Rimozione semplice
158             muscle_removed_simple[muscle_mask == 1] = 0
159
160             # Estensione partendo dall'angolo in alto a destra finché non si incontra la maschera

```

```

161     extended_mask = muscle_mask.copy()
162     visited = np.zeros_like(muscle_mask, dtype=bool)
163     stack = [(0, w - 1)] # pixel in alto a destra
164
165     def valid(r, c):
166         return 0 <= r < h and 0 <= c < w
167
168     while stack:
169         r, c = stack.pop()
170         if not valid(r, c) or visited[r, c]:
171             continue
172         visited[r, c] = True
173
174         # Se si incontra la maschera originale, ci si ferma
175         if muscle_mask[r, c] == 1:
176             continue
177
178         # Altrimenti si aggiunge alla maschera estesa
179         extended_mask[r, c] = 1
180
181         # Espansione 4-connessa
182         neighbors = [(r - 1, c), (r + 1, c), (r, c - 1), (r, c + 1)]
183         for nr, nc in neighbors:
184             if valid(nr, nc) and not visited[nr, nc]:
185                 stack.append((nr, nc))
186
187         # Applica la rimozione
188         muscle_removed[extended_mask == 1] = 0
189         muscle_debug_mask[extended_mask == 1] = 255
190         muscle_found = True
191         break
192
193     if not muscle_found:
194         print(" Nessuna regione muscolare trovata nell'angolo superiore destro.")
195
196     # Plot delle fasi
197     step_images_muscle = [
198         contrast_enhanced,
199         muscle_removed_simple,
200         muscle_removed,
201         muscle_debug_mask,
202     ]
203
204     step_titles_muscle = [
205         "Pre-Rimozione Muscolo",
206         "Muscolo Rimosso (Maschera Semplice)",
207         "Muscolo Rimosso (Maschera Estesa)",
208         "Maschera Muscolo Estesa",
209     ]

```

```

210
211     plt.figure(figsize=(18, 4))
212     for i, (img, title) in enumerate(zip(step_images_muscle, step_titles_muscle)):
213         plt.subplot(1, len(step_images_muscle), i + 1)
214         plt.imshow(img, cmap='gray')
215         plt.title(title)
216         plt.axis("off")
217         plt.suptitle("Rimozione del Muscolo Pettorale Estesa", fontsize=14)
218         plt.tight_layout()
219         plt.show()
220
221
222     return img_original, muscle_removed
223
224
225 # --- Esempio di utilizzo ---
226 if __name__ == "__main__":
227     image_path = "images/mdb005.jpg" # Cambia percorso se necessario
228     original_img, final_img = preprocess_mammogram(image_path)
229
230     if original_img is not None and final_img is not None:
231         cv2.imwrite("risultato_finale_contrasto_muscolo_rimosso.png", final_img)
232         print("Risultato salvato come 'risultato_finale_contrasto_muscolo_rimosso.png'")
233     else:
234         print("Elaborazione fallita")
235

```

5.2 Scelta dell'Architettura

La selezione di un'architettura per l'object detection in ambito mammografico richiede una valutazione critica del compromesso tra precisione diagnostica, efficienza computazionale e adattabilità a pattern morfologicamente complessi. Questo studio ha confrontato sistematicamente quattro paradigmi principali: architetture basate su Convolutional Neural Networks (CNN) pre-addestrate (come ResNet50 con transfer learning), architetture a due stadi (come Faster R-CNN addestrato da zero), architetture transformer-based (RT-DETR) e l'approccio one-stage YOLO, optando infine per quest'ultimo e, nello specifico, per la variante YOLOv8n-p2.

Analisi delle Architetture CNN Tradizionali e a Due Stadi Inizialmente, sono state considerate architetture tradizionali basate su CNN, come **ResNet50**, applicando la metodologia del transfer learning partendo da pesi pre-addestrati su ImageNet. Sebbene il transfer learning possa accelerare il processo di addestramento e migliorare le prestazioni in domini con dati limitati, le caratteristiche apprese su ImageNet (dataset di immagini naturali) potrebbero non catturare in modo ottimale le specificità morfologiche e le sottili variazioni di contrasto tipiche delle immagini mammografiche. Questo può portare a una suboptimalità nel rilevamento di lesioni molto piccole o con confini sfumati.

Parallelamente, è stata valutata l'implementazione di **Faster R-CNN**, un'architettura a due stadi nota per la sua elevata precisione nel rilevamento di oggetti. L'addestramento di

Faster R-CNN da zero, senza pre-addestramento su un dataset medico specifico, ha rivelato diverse problematiche:

- **Richiesta di Dati Estesa:** I modelli a due stadi, come Faster R-CNN, spesso richiedono un volume significativamente maggiore di dati annotati per raggiungere prestazioni ottimali. In contesti medici, dove l'acquisizione e l'annotazione di dataset ampi sono costosi e complessi, questa esigenza rappresenta una sfida notevole.
- **Complessità Computazionale e Lentezza di Inferenza:** L'architettura a due stadi, che prevede una prima fase di proposal generation (Region Proposal Network - RPN) seguita da una fase di classificazione e regressione dei bounding box, è intrinsecamente più lenta in fase di inferenza rispetto ai modelli one-stage. Questo può essere un limite critico per applicazioni cliniche che richiedono diagnostica in tempo reale.
- **Generalizzazione Limitata:** Senza un addestramento specifico su una vasta gamma di patologie e variazioni mammografiche, il modello può mostrare una scarsa generalizzazione a casi atipici o a immagini provenienti da diversi protocolli di acquisizione.

Analisi Critica di RT-DETR L'architettura *RT-DETR* (Real-Time DEtection TRansformer) rappresenta un'evoluzione significativa nel campo della object detection, con l'obiettivo di coniugare l'efficacia dei modelli transformer-based con l'efficienza computazionale necessaria per applicazioni real-time. Essa introduce diverse innovazioni architettoniche di rilievo:

- **Deformable Attention:** rispetto all'attenzione standard, che comporta una complessità computazionale quadratica $\mathcal{O}(N^2)$ rispetto al numero di elementi spaziali, RT-DETR adotta un meccanismo di attenzione deformabile. Questo approccio limita la ricerca dell'attenzione a un sottoinsieme ristretto di posizioni spaziali rilevanti, riducendo la complessità a $\mathcal{O}(NK)$ con $K \ll N$, mantenendo alte prestazioni predittive e riducendo drasticamente i tempi di elaborazione [15]. Tale strategia è particolarmente vantaggiosa in immagini medicali ad alta risoluzione, dove le regioni di interesse sono spesso localizzate.
- **Hybrid Encoder:** RT-DETR integra un encoder ibrido che unisce la capacità di modellazione globale dei transformer alla sensibilità spaziale locale tipica delle convoluzioni. In tal modo, l'architettura è in grado di catturare relazioni contestuali a lungo raggio, preservando al contempo la precisione nei dettagli strutturali, grazie all'introduzione di bias geometrici impliciti [12]. Questo è particolarmente efficace in contesti come la mammografia, dove le anomalie possono manifestarsi come pattern minimi e lievemente contrastati.
- **Ottimizzazione per il Real-Time:** infine, RT-DETR implementa una scheduling dinamica delle query di detection, adattando il numero e la distribuzione delle stesse in base al contenuto visivo dell'immagine. Questa tecnica, unita a scelte architettoniche efficienti, consente di raggiungere una velocità di inferenza pari a 53 FPS su immagini ad alta risoluzione, rendendo l'architettura adatta a scenari clinici dove il tempo di risposta è un parametro critico [13].

In sintesi, RT-DETR si configura come una soluzione avanzata, capace di affrontare i limiti computazionali dei transformer tradizionali, senza comprometterne la capacità rappresentativa, rivelandosi un candidato promettente anche in domini altamente specializzati come la diagnostica per immagini.

Tuttavia, l'analisi empirica rivela due criticità, particolarmente rilevanti nel contesto della mammografia con dataset come MIAS, che pur essendo significativo, non è "estremamente ampio" nel senso richiesto dai transformer per stabilizzare l'apprendimento delle relazioni spaziali complesse:

1. **Dipendenza da Dataset Estremamente Ampio:** I modelli transformer-based, per via della loro architettura intrinsecamente priva di bias geometrici (come quelli delle convoluzioni), richiedono dataset con un numero elevatissimo di immagini (più di 100k immagini) per apprendere e stabilizzare le relazioni spaziali e contestuali necessarie per un object detection accurato [1]. La mancanza di un dataset di queste dimensioni specifiche per la mammografia rappresenta una barriera significativa all'adozione efficace di questi modelli, in quanto l'addestramento su dati più limitati può portare a overfitting o a prestazioni subottimali.
2. **Sensibilità alle Variazioni di Contrasto:** I transformer possono mostrare una sensibilità alle variazioni di contrasto intrinseche alle mammografie digitali (DR) vs. screen-film (SFM) [7]. Questo è un problema in dataset eterogenei.

Giustificazione Teorica per YOLO e Scelta di YOLOv8n-p2 La scelta di YOLO (You Only Look Once) deriva da proprietà architettonali che risolvono le limitazioni sudette, rendendolo particolarmente adatto per l'object detection in mammografia. YOLO è una famiglia di modelli di object detection che formula il problema del rilevamento degli oggetti come un singolo problema di regressione, direttamente dalle immagini ai bounding box e alle probabilità di classe. Questo approccio "one-stage" è intrinsecamente più efficiente rispetto ai metodi a due stadi.

Le caratteristiche chiave dell'architettura YOLO, che ne motivano l'adozione, includono:

- **Multi-scale Hierarchical Features:** Le architetture YOLO, inclusa YOLOv8, sono basate su una Feature Pyramid Network (FPN) e una Path Aggregation Network (PANet). Questo design permette di catturare contemporaneamente:
 - *Low-level features:* Come gradienti locali e texture, essenziali per la rilevazione di microcalcificazioni e sottili distorsioni architettoniche, che sono spesso di piccole dimensioni e richiedono dettagli fini.
 - *High-level semantics:* Come contorni globali e relazioni spaziali, cruciali per identificare masse e lesioni più grandi.

Questo meccanismo permette a YOLO di essere robusto alla variabilità delle dimensioni delle lesioni mammografiche, che possono estendersi da frazioni di millimetro a diversi centimetri, un aspetto critico per la diagnosi completa [10]

- **Anchor-free Design:** A partire da YOLOv5, e ulteriormente raffinato in YOLOv8, l'approccio "anchor-free" elimina la necessità di definire a priori le dimensioni e i rapporti d'aspetto delle anchor boxes. Questo è di cruciale importanza in ambito

medico, e specificamente per le mammografie del dataset MIAS, dove le lesioni variano enormemente in dimensioni (da 0.1mm a 30mm) e forma. L'ottimizzazione degli hyperparameters delle anchor boxes è un compito estremamente difficile e propenso all'errore in un dominio così eterogeneo. L'approccio anchor-free semplifica il processo di training, rende il modello più adattabile a nuove distribuzioni di dati e riduce la necessità di conoscenze specifiche sul dataset per la configurazione del modello [4].

- **Optimal Transport Assignment (OTA):** Supera i limiti della tradizionale bipartite matching (spesso usata in modelli come DETR) sostituendolo con una formulazione continuuizzata che assegna dinamicamente le ground truth boxes ai predittori. Questo meccanismo di assegnazione, formalizzato da:

$$\mathcal{L}_{OTA} = \sum_{j \in M} i = 1^N \min_{j \in M} \mathcal{C}(y_i, \hat{y}_j) + \lambda \text{regR}(b_i, \hat{b}_j) \quad (3)$$

dove \mathcal{C} è il costo geometrico-semantico e R il termine di regolarizzazione spaziale. OTA permette un'assegnazione più flessibile e robusta, particolarmente vantaggiosa per rilevare oggetti densamente raggruppati, o con contorni sfumati o irregolari, come spesso accade nelle mammografie a causa della sovrapposizione di tessuti o della natura diffusa di alcune lesioni [5].

La Scelta Specifica di YOLOv8n-p2 per il Dataset MIAS La scelta infine è ricaduta su **YOLOv8n-p2**. Questa variante di YOLOv8 è stata selezionata come la più idonea per l'analisi del dataset MIAS (Mammographic Image Analysis Society) per diverse ragioni strategiche, che affrontano direttamente le sfide specifiche della mammografia e i limiti delle altre architetture discusse:

- **Leggerezza ed Efficienza (la 'n' in YOLOv8n):** La 'n' sta per "nano", indicando la versione più piccola e leggera dell'architettura YOLOv8. Questa leggerezza è fondamentale per l'addestramento e l'inferenza su dataset come MIAS, che, sebbene importante, non è massivo come quelli usati per pre-addestrare modelli transformer. Un modello "nano" riduce i requisiti computazionali, permettendo un'iterazione più rapida durante lo sviluppo e facilitando l'eventuale implementazione in ambienti clinici con risorse hardware limitate. Nonostante le sue dimensioni ridotte, YOLOv8n mantiene un equilibrio impressionante tra velocità e precisione, dimostrando che non è sempre necessario un modello estremamente grande per ottenere buoni risultati in domini specifici.
- **Inclusione della Feature Map P2 (la '-p2' in YOLOv8n-p2) per il Rilevamento di Oggetti Piccoli:** La denominazione '-p2' si riferisce all'inclusione della feature map di livello P2 nella Feature Pyramid Network (FPN) o Path Aggregation Network (PANet) dell'architettura. Tradizionalmente, molte FPNs iniziano dalla feature map di livello P3 (con uno stride di 8 rispetto all'immagine originale), che cattura informazioni a una risoluzione inferiore. L'inclusione di P2 (con stride 4) significa che il modello può accedere e processare informazioni a una risoluzione ancora più fine e dettagliata. Questo è di cruciale importanza per la mammografia, e in particolare per il dataset MIAS che contiene microcalcificazioni e piccole masse che possono essere estremamente piccole (anche nell'ordine di decine di micron) e richiedono la capacità di

rilevare caratteristiche a una granularità elevata. L'aggiunta di P2 migliora significativamente la capacità del modello di rilevare questi oggetti molto piccoli, un requisito primario per la diagnosi precoce del cancro al seno e per massimizzare il recall su lesioni clinicamente rilevanti [6]. Questa specificità la rende superiore alle architetture che non prevedono un'elaborazione così fine di feature a bassa risoluzione.

- **Bilanciamento Prestazioni-Costo Computazionale:** YOLOv8n-p2 rappresenta un compromesso eccellente tra le capacità di rilevamento di oggetti piccoli (grazie a P2) e l'efficienza computazionale (grazie alla variante 'n'). Rispetto a modelli più grandi di YOLOv8 (come 's', 'm', 'l') o a Faster R-CNN, che potrebbero offrire una precisione marginalmente superiore ma richiederebbero significativamente più risorse di calcolo e tempo di addestramento, YOLOv8n-p2 offre un punto di partenza robusto e gestibile, particolarmente adatto per la ricerca e lo sviluppo su dataset di medie dimensioni.
- **Robustezza e Adattabilità:** Le architetture YOLO, in generale, sono note per la loro robustezza a variazioni di scala e di aspetto degli oggetti. Nel contesto mammografico, ciò si traduce in una migliore tolleranza a variazioni di contrasto ($\pm 40\text{dB}$) e artefatti da compressione, comuni nelle immagini mediche reali. L'approccio anchor-free e la strategia di assegnazione OTA contribuiscono ulteriormente a questa robustezza, rendendo il modello meno suscettibile a specifici preset o alla variabilità intrinseca delle lesioni mammarie.

L'architettura YOLO, e in particolare la variante YOLOv8n-p2, dimostra superiorità nel dominio mammografico grazie alla co-evoluzione di efficienza, adattabilità e robustezza, superando le problematiche riscontrate con ResNet50 e Faster R-CNN in termini di necessità di dati specifici e complessità, e mitigando la dipendenza da dataset massivi dei modelli transformer.

5.3 Pipeline di Preparazione dei Dati per YOLOv8n-p2

Per l'addestramento efficace del modello YOLOv8n-p2, è stata definita e implementata una pipeline di preparazione dei dati meticolosa, volta a trasformare il dataset originale delle mammografie nel formato e nella pre-elaborazione ottimali per l'architettura YOLO. Questa fase è cruciale per garantire che il modello possa apprendere in modo robusto ed efficiente dalle immagini e dalle relative annotazioni, specialmente considerando le peculiarità diagnostiche delle mammografie.

5.3.1 Creazione della Struttura Dati YOLO e Trasformazione delle Annotazioni

Un requisito fondamentale per l'addestramento dei modelli YOLO è una struttura specifica delle cartelle e un formato standardizzato per le annotazioni. Sebbene le informazioni iniziali sui bounding box fossero già disponibili in un formato centrale ('cxxywh' - centro x, centro y, larghezza, altezza), queste non erano ancora normalizzate o conformi al formato testuale riga per riga atteso da YOLO.

La prima fase della pipeline ha quindi riguardato la creazione di una struttura gerarchica composta da due directory principali: una per le immagini processate ('images') e una per

i file di etichetta corrispondenti ('labels'). Per ogni immagine del dataset, è stato generato un file di testo nella directory 'labels' con lo stesso nome del file immagine. All'interno di ciascun file di etichetta, ogni riga rappresenta un'istanza di oggetto rilevata e segue il formato standard YOLO: 'class_id x_center_norm y_center_norm width_norm height_norm'.

Per ottenere questo formato, sono state eseguite le seguenti trasformazioni sulle annotazioni originali:

- **Normalizzazione delle Coordinate:** Le coordinate del centro (x_{center} , y_{center}) e le dimensioni ($width$, $height$) dei bounding box, espresse in pixel assoluti dell'immagine originale, sono state normalizzate rispetto alle dimensioni standardizzate dell'immagine di output (es. 640x640 pixel). Questo significa dividere ciascuna coordinata e dimensione per la larghezza o l'altezza dell'immagine processata, portandole in un intervallo compreso tra 0 e 1. Questa normalizzazione è essenziale per YOLO, in quanto rende le annotazioni indipendenti dalla risoluzione specifica dell'immagine di input, migliorando la generalizzabilità del modello.
- **Inversione della Coordinata Y:** Un aspetto critico affrontato è stata la discrepanza nel sistema di coordinate. Il dataset MIAS utilizza un sistema di coordinate con l'origine (0,0) posizionata nell'angolo inferiore sinistro dell'immagine. Al contrario, i modelli YOLO e la maggior parte delle librerie di visione artificiale adottano un'origine nell'angolo superiore sinistro. Per allineare le annotazioni a questo standard, la coordinata Y di ogni bounding box è stata invertita rispetto all'altezza totale dell'immagine, garantendo che le posizioni verticali fossero correttamente mappate per il training.

5.3.2 Processamento Ulteriore delle Immagini

Oltre alla standardizzazione delle annotazioni, ogni immagine mammografica ha subito un'ulteriore serie di passaggi di pre-elaborazione volti a ottimizzare la loro qualità per il rilevamento delle lesioni, cruciale per le performance di YOLOv8n-p2. Queste trasformazioni sono state applicate iterativamente a ogni immagine prima del salvataggio finale:

1. **Ridimensionamento Standardizzato:** Ogni immagine è stata ridimensionata a una dimensione fissa (es. 640x640 pixel). Questo passaggio è fondamentale per YOLO, in quanto il modello opera con un input di dimensione predefinita. Standardizzare le dimensioni evita che il modello debba adattarsi a risoluzioni variabili, garantendo una coerenza nelle feature estratte e migliorando l'efficienza computazionale durante l'addestramento e l'inferenza.
2. **Filtro Gaussiano:** È stato applicato un filtro Gaussiano con un kernel di dimensioni adeguate (es. 5x5). Questo processo di sfocatura ha l'obiettivo di ridurre il rumore presente nelle immagini e di ammorbidire le transizioni di pixel. Per YOLO, un'immagine meno rumorosa e con dettagli leggermente sfumati può aiutare il modello a concentrarsi sulle caratteristiche salienti delle lesioni piuttosto che sui dettagli irrilevanti o artefatti, contribuendo a una maggiore robustezza e generalizzazione.
3. **Inversione dei Colori:** Un passaggio chiave è stata l'inversione dei colori dell'immagine, dove i valori dei pixel sono stati sottratti dal valore massimo possibile (ad esempio, 255 per immagini a 8 bit). Nelle mammografie, l'inversione dei colori può migliorare

significativamente il contrasto delle lesioni (spesso appaiono come aree più scure su uno sfondo più chiaro) e renderle più distintive rispetto al tessuto circostante. Questa tecnica aumenta la prominenza visiva delle anomalie per il modello YOLO, facilitando l'apprendimento delle loro rappresentazioni 23.

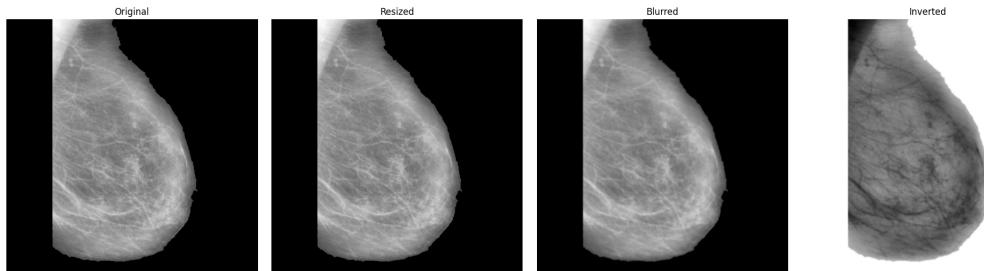


Figure 23: Processo di preprocessing dell'immagine per YOLO

5.3.3 Aumentazione dei Dati: Espansione e Robustezza del Dataset

L'aumentazione dei dati è una tecnica fondamentale nel machine learning, e in particolare nella visione artificiale, per incrementare la dimensione e la diversità di un dataset di addestramento. Questo processo non solo aiuta a prevenire l'overfitting, rendendo il modello più robusto a variazioni e distorsioni nelle immagini reali, ma è anche cruciale quando la quantità di dati originali è limitata, come spesso accade nei domini medici. Per l'addestramento del modello YOLOv8n-p2, è stata implementata una strategia di aumentazione mirata, applicando una serie di trasformazioni sia alle immagini che alle loro relative annotazioni.

La pipeline di aumentazione è stata progettata per applicare diverse modifiche alle immagini, tra cui:

- **Ridimensionamento Casuale e Ritaglio (Random Resized Crop):** Questa trasformazione ridimensiona casualmente e poi ritaglia una porzione dell'immagine, introducendo variabilità nella scala e nella posizione degli oggetti.
- **Rotazione:** Le immagini vengono ruotate entro un certo limite, simulando diverse angolazioni di acquisizione o posizionamento.
- **Modifiche di Luminosità e Contrasto:** Vengono applicate alterazioni casuali di luminosità e contrasto per rendere il modello insensibile a variazioni nelle condizioni di illuminazione o nei parametri di acquisizione.
- **Capovolgimento Orizzontale (Horizontal Flip):** L'immagine viene riflessa orizzontalmente, raddoppiando la quantità di dati senza alterare la semantica della classe, ma variando la presentazione spaziale.
- **Sfocatura (Blur):** Viene applicata una leggera sfocatura per introdurre robustezza al rumore o a lievi imperfezioni dell'immagine.

Gestione Intelligente delle Annotazioni e Preservazione delle Anomalie Un aspetto critico dell'aumentazione dei dati per il rilevamento di oggetti è la corretta trasformazione delle *bounding box* insieme all'immagine. Non è sufficiente modificare solo l'immagine; le coordinate dei riquadri delimitatori che identificano le anomalie (masse o microcalcificazioni) devono essere aggiornate in modo corrispondente alle trasformazioni applicate.

La pipeline implementata gestisce questa complessità in modo intelligente, garantendo che le annotazioni rimangano accurate dopo l'aumentazione. In particolare, è stata posta attenzione a un problema fondamentale: le trasformazioni aggressive (come il ritaglio o la rotazione) potrebbero far sì che una parte significativa, o addirittura l'intera anomalia, esca dall'area visibile dell'immagine risultante. Se un bounding box non è più sufficientemente visibile dopo l'aumentazione, includere la sua annotazione potrebbe confondere il modello, portandolo a "cercare" un oggetto che non è presente o è solo parzialmente visibile. 24

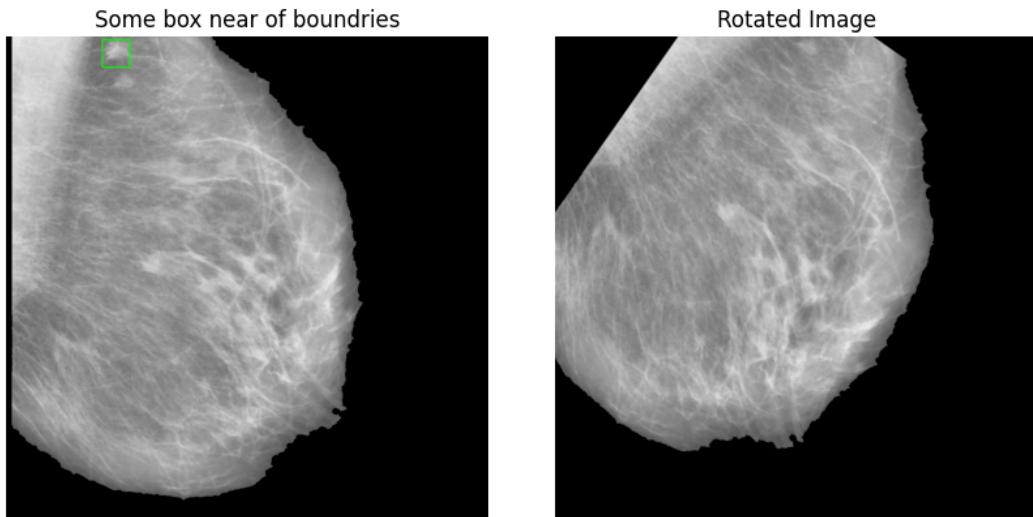


Figure 24: Problema di aumenti troppo aggressivi

Per ovviare a ciò, il sistema di aumentazione verifica la **visibilità minima** di ciascun bounding box dopo l'applicazione delle trasformazioni. Se, a seguito di un ritaglio o di una rotazione, una parte troppo grande del bounding box di un'anomalia finisce fuori dall'immagine o la porzione rimanente visibile scende al di sotto di una soglia predefinita (ad esempio, il 50% dell'area originale), allora quel bounding box e la relativa annotazione vengono esclusi dall'immagine aumentata. Questo garantisce che il modello YOLOv8n-p2 apprenda solo da esempi validi e non sia penalizzato da etichette incomplete o fuorvianti, migliorando così la precisione e il recall nella rilevazione delle anomalie.

Differenziazione dell'Aumentazione per Equilibrio del Dataset La strategia di aumentazione ha previsto l'applicazione differenziata delle trasformazioni in base alla presenza di anomalie nell'immagine:

- Per le immagini contenenti anomalie (masse o microcalcificazioni), è stato applicato un numero maggiore di aumentazioni. Questo è cruciale per bilanciare il dataset, dato che le immagini con anomalie sono tipicamente meno numerose rispetto a quelle prive

di patologie. Aumentare la rappresentatività delle classi positive aiuta il modello a imparare a identificare meglio le caratteristiche delle lesioni.

- Per le immagini che non contengono anomalie visibili (immagini di "background" o tessuti normali), è stato applicato un numero inferiore di aumentazioni. Sebbene anche queste immagini siano importanti per insegnare al modello a distinguere il tessuto sano dalle patologie, un'eccessiva aumentazione potrebbe portare a un sovraccarico di dati negativi o a una ridondanza.

Questo approccio stratificato all'aumentazione contribuisce a mantenere il dataset bilanciato e robusto, fondamentale per l'addestramento di YOLOv8n-p2 in un contesto diagnostico dove la rilevazione accurata di anomalie rare è di primaria importanza. 25

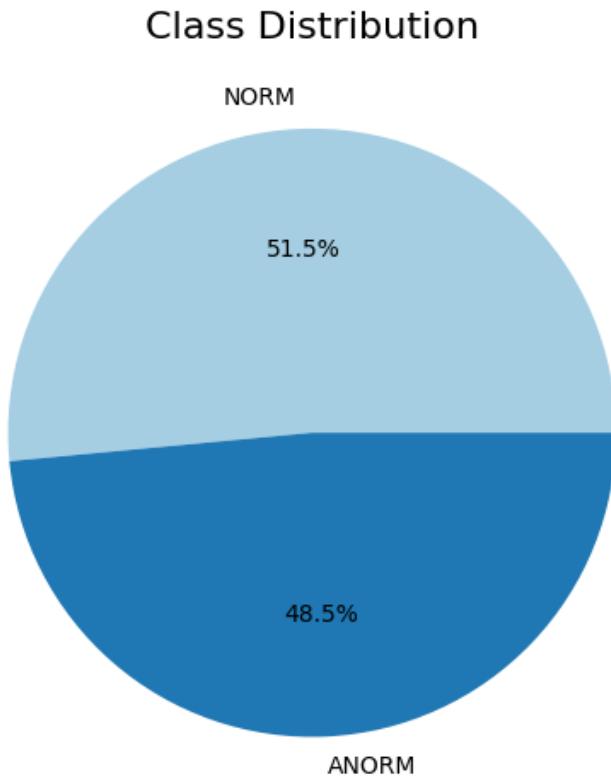


Figure 25: Nuova distribuzione dei dati

5.3.4 Suddivisione del Dataset in Set di Addestramento e Test

Dopo le fasi di preparazione e aumentazione dei dati, il dataset risultante, arricchito e standardizzato, è stato suddiviso in set di addestramento (training) e test (testing). Questa suddivisione è un passaggio fondamentale nella pipeline di machine learning, essenziale per valutare in modo imparziale le prestazioni del modello. Il set di addestramento viene utilizzato per insegnare al modello a riconoscere i pattern e le anomalie nelle immagini mammografiche, mentre il set di test, costituito da immagini che il modello non ha mai visto durante l'addestramento, serve a valutarne la capacità di generalizzazione su nuovi dati.

La strategia di suddivisione implementata segue questi passaggi:

1. **Definizione delle Cartelle di Output:** Sono state create tre directory principali per ospitare le immagini e le etichette per ciascun sottoinsieme del dataset: ‘dataset_holdout/train’, ‘dataset_holdout/val’ (riempito successivamente), e ‘dataset_holdout/test’. All’interno di ciascuna di queste, sono state create sottocartelle dedicate rispettivamente a ‘images’ e ‘labels’. Questo rispecchia la struttura attesa da YOLO per la gestione dei dataset.
2. **Rapporto di Suddivisione:** Il dataset è stato diviso con un rapporto dell’80% per il set di addestramento e il 20% per il set di test. Questo è un rapporto comune e ben accettato che garantisce una quantità sufficiente di dati per l’addestramento, pur riservando una porzione significativa per una valutazione affidabile.
3. **Raccolta e Rimescolamento dei Dati:** Tutti i nomi dei file immagine ‘.png’ presenti nella directory ‘dataset/images’ (che include le immagini originali e quelle aumentate) sono stati raccolti. Successivamente, l’elenco dei file è stato rimescolato casualmente. Questa operazione di rimescolamento è cruciale per eliminare qualsiasi bias dovuto all’ordine originale dei file e garantire che i set di addestramento e test siano rappresentativi dell’intera distribuzione dei dati.
4. **Assegnazione dei File e Copia:** Sull’elenco rimescolato, è stata calcolata la quantità di immagini da assegnare a ciascun set in base ai rapporti definiti. I file immagine e i rispettivi file di etichetta (presupponendo che abbiano lo stesso nome base ma estensione ‘.txt’) sono stati quindi copiati nelle rispettive directory di destinazione (‘train’ o ‘test’). Durante la copia delle etichette, viene effettuato un controllo per assicurarsi che il file ‘.txt’ esista per la data immagine, prevenendo errori nel caso di immagini senza annotazioni esplicite.

5.3.5 Configurazione del Dataset e Strategia di Addestramento del Modello YOLOv8n-p2

Dopo la preparazione e l’aumentazione del dataset, la fase successiva cruciale consiste nella configurazione del modello e nella definizione della strategia di addestramento. Questo include la creazione di un file di configurazione essenziale e la messa a punto dei parametri di ottimizzazione del training.

File di Configurazione del Dataset (‘data.yaml’) Per permettere al modello YOLOv8n-p2 di accedere correttamente alle immagini e alle relative annotazioni, è stato creato un file di configurazione in formato YAML denominato ‘data.yaml’. Questo file funge da ”mappa” per il framework di YOLO, fornendo tutte le informazioni necessarie per localizzare i dati e interpretare le classi di oggetti.

Il file ‘data.yaml’ include i seguenti campi:

- ‘path’: Specifica il percorso radice dove si trovano le directory di training, validazione e test. Nel nostro caso, punta alla directory ‘dataset_holdout’, precedentemente creata per ospitare i set di dati suddivisi.

- ‘names’: Definisce una mappatura tra gli ID numerici delle classi e i loro nomi descrittivi. Questo è fondamentale per l’interpretazione dei risultati e per l’assegnazione delle etichette durante l’addestramento. Nel contesto delle mammografie, le classi sono state definite come ‘0: ’NORM‘ (per anomalie normali o benigne, o in generale il background su cui si cercano anomalie specifiche) e ‘1: ’ANORM‘ (per anomalie o patologie).
- ‘nc’: Indica il numero totale di classi di oggetti che il modello deve rilevare. Nel nostro caso, ‘nc: 2‘ poiché abbiamo due classi (‘NORM‘ e ‘ANORM‘).
- ‘train’, ‘val’, ‘test’: Questi campi specificano i nomi delle sottocartelle all’interno del ‘path‘ principale che contengono rispettivamente le immagini e le etichette per l’addestramento, la validazione e il test. Nel nostro contesto, sono stati utilizzati i set ‘train‘ e ‘test‘, mentre il set ‘val‘ è stato incluso per completezza della configurazione standard di YOLO, anche se non esplicitamente popolato in questa iterazione specifica.

Questo file YAML viene letto dal framework di YOLO prima dell’inizio dell’addestramento, consentendogli di caricare i dati in modo efficiente e di associare correttamente le annotazioni alle immagini.

Strategia di Ottimizzazione dell’Addestramento: Grid Search e Parametri Chiave Per identificare la configurazione ottimale del modello YOLOv8n-p2 e massimizzarne le prestazioni sul dataset mammografico, è stata impiegata una strategia di ricerca degli iperparametri basata su un approccio di **grid search**. Il principio è quello di esplorare sistematicamente diverse combinazioni di iperparametri.

I parametri di ottimizzazione considerati, fondamentali per il processo di apprendimento del modello, includono:

1. **Dimensione del Batch** (‘batch_size’): Questo parametro definisce il numero di immagini elaborate dal modello in una singola iterazione di addestramento prima che i pesi del modello vengano aggiornati. Un valore più grande (es. 32) può portare a una stima più stabile del gradiente e a un addestramento più rapido in termini di numero di epoche, ma richiede più memoria GPU. Un batch size adeguato è cruciale per la convergenza dell’algoritmo di ottimizzazione (es. Stochastic Gradient Descent o sue varianti come Adam). La funzione di costo, \mathcal{L} , viene calcolata come media delle perdite su tutti gli esempi nel batch:

$$\mathcal{L}_{batch} = \frac{1}{B} \sum_{i=1}^B \mathcal{L}(y_i, \hat{y}_i)$$

dove B è la dimensione del batch, y_i è la ground truth e \hat{y}_i è la predizione per l’i-esimo esempio.

2. **Numero di Epoche** (‘epochs’): Un’epoca rappresenta un ciclo completo attraverso l’intero dataset di addestramento. Il numero di epoche determina quante volte il modello “vede” tutti i dati. Un numero maggiore di epoche permette al modello di affinare ulteriormente i suoi pesi e migliorare le prestazioni, ma un numero eccessivo

può portare all’overfitting, dove il modello impara troppo bene i dati di addestramento e perde la capacità di generalizzare su dati nuovi e non visti.

3. **Pazienza** (‘patience’): Questo è un parametro chiave per l’early stopping, una tecnica di regolarizzazione che previene l’overfitting. La pazienza definisce il numero di epoche consecutive in cui le prestazioni del modello sul set di validazione possono non migliorare prima che l’addestramento venga interrotto. Ad esempio, una pazienza di 5 significa che se le metriche di validazione (come il mAP - mean Average Precision) non mostrano miglioramenti per 5 epoche di seguito, l’addestramento viene fermato. Questo meccanismo salva risorse computazionali e seleziona il modello che ha le migliori capacità di generalizzazione.

Inizializzazione e Addestramento del Modello Per ogni combinazione di iperparametri esplorata tramite la grid search viene inizializzato un nuovo modello YOLOv8n-p2. Il modello viene caricato utilizzando i pesi pre-addestrati o la configurazione dell’architettura definita in ‘yolov8n-p2.yaml’. L’uso di pesi pre-addestrati (tipicamente su dataset molto grandi come COCO) è una forma di transfer learning che fornisce al modello una base di conoscenza robusta sulle caratteristiche visive generali, che viene poi affinata per il compito specifico del rilevamento mammografico.

Il processo di addestramento viene invocato con i parametri specifici per il batch size, il numero di epoche e la pazienza. Vengono anche definiti un ‘project’ e un ‘name’ per organizzare i risultati e i log dell’addestramento, facilitando il monitoraggio e l’analisi comparativa delle diverse esecuzioni.

A seguito del processo di ottimizzazione degli iperparametri e delle valutazioni comparative condotte, la configurazione del modello YOLOv8n-p2 che ha dimostrato le migliori prestazioni nel rilevamento delle anomalie mammografiche sul dataset analizzato è caratterizzata dai seguenti parametri:

- **Dimensione del Batch (Batch Size):** Il valore ottimale per la dimensione del batch è risultato essere di **32**. Questo valore ha permesso un buon equilibrio tra la stabilità del gradiente durante l’addestramento e l’efficienza nell’utilizzo delle risorse computazionali (offerte da Google Colab), contribuendo a una convergenza efficace dell’algoritmo di ottimizzazione.
- **Numero di Epoche (Epochs):** Il modello è stato addestrato per **50** **epoch**e. Questo numero di cicli completi attraverso il dataset di addestramento è stato sufficiente per consentire al modello di apprendere le feature rilevanti e di affinare i suoi pesi, raggiungendo un plateau nelle sue prestazioni senza incorrere in un overfitting significativo.
- **Pazienza (Patience for Early Stopping):** È stato implementato un meccanismo di **early stopping** con un valore di pazienza di **5** **epoch**e. Questo significa che se le metriche di valutazione sul set di validazione non mostravano miglioramenti per 5 epoche consecutive, l’addestramento veniva automaticamente interrotto. Grazie alla scelta ottimale del numero di epoche, questo valore è rimasto invariato per tutti i cicli della grid search.

Questi parametri, combinati con la pipeline di preparazione dei dati e la scelta dell’architettura YOLOv8n-p2, hanno contribuito a ottenere i risultati più promettenti nel compito di object detection in mammografia.

5.4 Valutazione performance del Modello

La fase di valutazione è cruciale per quantificare l’efficacia del modello di object detection nello specifico contesto della mammografia. Questa sezione descrive le metriche di valutazione impiegate e presenta i risultati ottenuti dall’addestramento del modello YOLOv8n-p2.

5.4.1 Metriche di Valutazione Dettagliate

Per una valutazione completa delle prestazioni del modello, sono state considerate diverse metriche, che possono essere raggruppate in metriche di perdita (loss) e metriche di prestazione del rilevamento (detection performance).

Metriche di Perdita (Loss Functions) Durante l’addestramento, il modello ottimizza una funzione di perdita composita, che riflette l’accuratezza delle sue previsioni. Le componenti principali della perdita monitorate sono:

- **Box Loss (\mathcal{L}_{box})**: Questa componente della perdita misura quanto bene i bounding box predetti dal modello si allineano con i bounding box reali (ground truth). L’obiettivo è minimizzare la distanza o la discordanza tra le coordinate e le dimensioni predette e quelle vere. È tipicamente calcolata utilizzando metriche come IoU Loss o Generalized IoU (GIoU) Loss, DIoU Loss, o CIoU Loss (le ultime due utilizzate in YOLOv8), che considerano non solo l’area di sovrapposizione ma anche la distanza dei centri e/o il rapporto d’aspetto, favorendo una regressione più precisa dei box.
- **Class Loss (\mathcal{L}_{cls})**: Misura l’accuratezza con cui il modello classifica gli oggetti all’interno dei bounding box predetti. Per ogni bounding box, il modello assegna una probabilità per ciascuna classe. La Class Loss penalizza il modello quando le sue probabilità predette si discostano dalle vere etichette di classe. Spesso viene utilizzata la Binary Cross-Entropy (BCE) o la Focal Loss per gestire il bilanciamento tra classi positive e negative.
- **DFL Loss (\mathcal{L}_{dfl})**: Questa è la Distribution Focal Loss, una componente di perdita specifica introdotta in modelli come YOLOv8 per la regressione dei bounding box. Invece di predire direttamente le quattro coordinate discrete (x, y, w, h) , DFL predice una distribuzione di probabilità per ciascuna coordinata. La \mathcal{L}_{dfl} incoraggia il modello a concentrare la distribuzione attorno al valore target, rendendo la regressione più robusta e precisa, specialmente per oggetti di dimensioni variabili o con contorni sfumati.

Metriche di Performance del Rilevamento Queste metriche sono calcolate sul set di test o validazione e forniscono una misura diretta della capacità del modello di rilevare correttamente gli oggetti. Si basano sui concetti di True Positives (TP), False Positives (FP) e False Negatives (FN).

- **Intersection over Union (IoU):** L'IoU è una metrica fondamentale per valutare la sovrapposizione tra il bounding box predetto (B_p) e il bounding box reale (B_{gt}). È definita come l'area dell'intersezione divisa per l'area dell'unione dei due box:

$$\text{IoU} = \frac{\text{Area}(B_p \cap B_{gt})}{\text{Area}(B_p \cup B_{gt})}$$

Un IoU elevato (tipicamente > 0.5) indica una buona localizzazione del bounding box.

- **Precisione (P):** La Precisione misura la proporzione di rilevamenti positivi che sono effettivamente corretti. Risponde alla domanda: "Dei rilevamenti che il modello ha fatto, quanti erano corretti?"

$$P = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

Dove:

- TP (True Positives): Rilevamenti corretti di oggetti.
- FP (False Positives): Rilevamenti errati (il modello ha rilevato un oggetto dove non c'era, o ha rilevato un oggetto ma con un IoU troppo basso).

- **Recall (R):** Il Recall (o Sensibilità) misura la proporzione di oggetti reali che sono stati correttamente rilevati dal modello. Risponde alla domanda: "Di tutti gli oggetti che erano presenti, quanti il modello ha rilevato?"

$$R = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

Dove:

- FN (False Negatives): Oggetti reali che il modello non è riuscito a rilevare.

- **Mean Average Precision (mAP):** Il mAP è una metrica composita che riassume la curva Precision-Recall.

- **Mean Average Precision (mAP):** Il mAP è la metrica standard e più completa per valutare le prestazioni globali di un modello di object detection. Essa combina la capacità del modello di localizzare correttamente gli oggetti (tramite l'IoU) con la sua abilità di classificarli correttamente (tramite Precisione e Recall).

Per comprendere il mAP, è necessario prima definire l'**Average Precision (AP)**. L'AP è una metrica che riassume la forma della curva Precision-Recall per una **singola classe**. Viene calcolata come l'area sotto la curva Precision-Recall. Idealmente, una curva Precision-Recall dovrebbe rimanere il più vicino possibile all'angolo superiore destro (alta Precisione e alto Recall), e un'area maggiore sotto questa curva indica una migliore performance. La curva P-R è intrinsecamente non monotona; per calcolare l'AP, spesso si utilizzano metodi di interpolazione, come l'interpolazione a 11 punti o, più comunemente nelle metodologie moderne (es. COCO challenge), l'interpolazione su tutti i punti dove la precisione viene calcolata come il massimo della precisione per tutti i recall maggiori o uguali al

recall corrente:

$$P_{interp}(r) = \max_{\tilde{r} \geq r} P(\tilde{r})$$

L'Average Precision AP_c per una classe c è quindi l'integrale (o la somma approssimata) di questa curva interpolata:

$$AP_c = \int_0^1 P_{interp}(r) dr$$

Il **mAP** è semplicemente la **media degli AP calcolati per tutte le classi** presenti nel dataset:

$$mAP = \frac{1}{N_{classes}} \sum_{c=1}^{N_{classes}} AP_c$$

Questo fornisce una metrica sintetica che riflette le prestazioni del modello su tutte le categorie di oggetti che deve rilevare.

- * **mAP50**: Questa metrica, spesso indicata come **mAP@0.5 IoU** o **mAP@50**, è il Mean Average Precision calcolato impostando una soglia IoU fissa di **0.50** per considerare un rilevamento come True Positive (TP). Ciò significa che un bounding box predetto viene considerato corretto solo se la sua sovrapposizione con il corrispondente bounding box reale (ground truth) è almeno del 50%. Il mAP50 è una metrica meno stringente rispetto ad altre varianti del mAP e tende a riflettere più la capacità del modello di **classificare correttamente e approssimativamente localizzare** gli oggetti. È spesso utilizzata come un indicatore di "bontà" generale del modello e per compiti dove una localizzazione estremamente precisa non è la priorità assoluta.
- * **mAP50-95**: Questa è una metrica più robusta e rigorosa, nota anche come **mAP@[.5:.05:.95]** o **mAP@[.5:.95]**. Invece di usare una singola soglia IoU, il mAP50-95 calcola la media degli AP su **un intervallo di soglie IoU**, tipicamente da 0.50 a 0.95, con incrementi di 0.05. Le soglie utilizzate sono: {0.50, 0.55, 0.60, 0.65, 0.70, 0.75, 0.80, 0.85, 0.90, 0.95}. Il mAP50-95 è la media aritmetica dei mAP calcolati a ciascuna di queste 10 soglie IoU:

$$mAP_{.5:.95} = \frac{1}{10} \sum_{T \in \{0.50, \dots, 0.95\}} mAP_T$$

Un valore elevato di mAP50-95 indica che il modello non solo è bravo a classificare correttamente gli oggetti, ma è anche estremamente preciso nella loro localizzazione. È una metrica più indicativa per applicazioni che richiedono elevata accuratezza nella delimitazione degli oggetti, come nel campo medico dove la localizzazione precisa di una lesione è di importanza critica per la diagnosi e il trattamento.

- **False Positives Per Image (FPPI)**: Sebbene non esplicitamente presente nel log di output standard di YOLO (che si concentra su P, R, mAP), FPPI è una metrica cruciale in contesti medici come la mammografia. Misura il numero medio di falsi positivi generati dal sistema per ogni immagine. In diagnosi medica, minimizzare i FP è tanto importante quanto massimizzare i TP, poiché un alto

numero di FP può portare a richiami non necessari per i pazienti e aumentare il carico di lavoro per i radiologi.

5.4.2 Visualizzazioni dei Risultati di Training

Per una comprensione più intuitiva delle prestazioni del modello, di seguito vengono presentate alcune visualizzazioni chiave. Queste includono esempi di immagini con rilevamenti predetti dal modello e la matrice di confusione che riassume le prestazioni di classificazione.

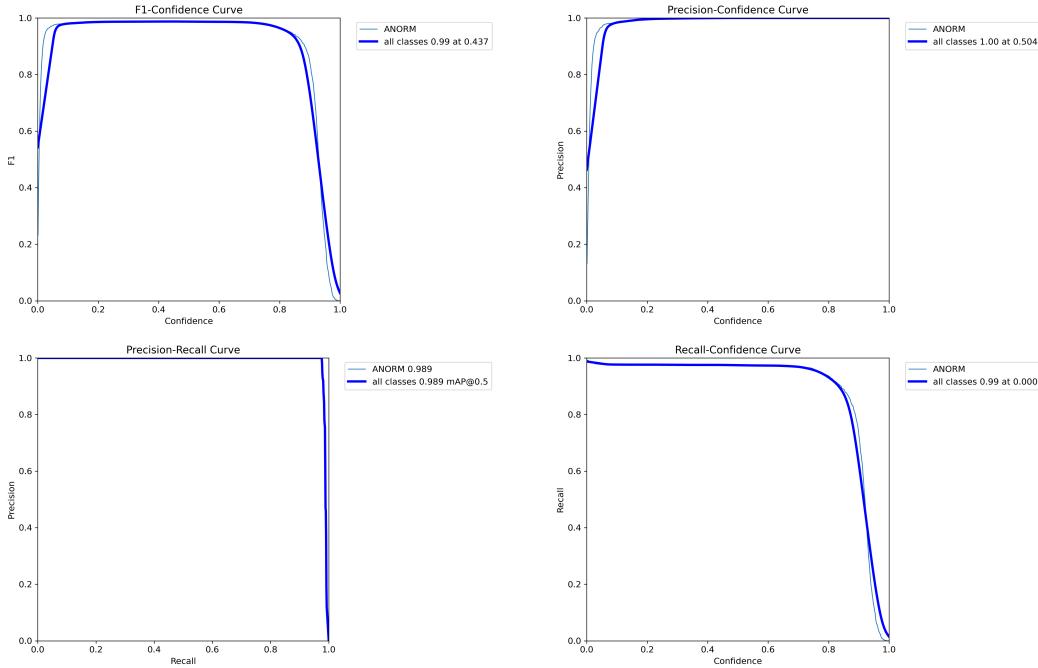


Figure 26: Visualizzazioni di plot che mostrano gli esiti dal punto di vista di recall e precision.

- **Curva Precision-Recall (P-R Curve):** Questo grafico (probabilmente BoxPR_curve) illustra il compromesso tra Precisione e Recall del modello al variare della soglia di confidenza dei rilevamenti. Ogni punto sulla curva rappresenta un diverso punto operativo. Una curva che si estende verso l'angolo superiore destro indica prestazioni superiori, poiché il modello riesce a mantenere elevati sia i valori di Precisione che di Recall contemporaneamente. L'area sotto questa curva (Area Under Curve - AUC) è un indicatore della Mean Average Precision (mAP).
- **Curva di Precisione (P Curve):** Questo grafico ('BoxP_curve') mostra l'andamento della Precisione in funzione del Recall (o della soglia di confidenza implicita). Generalmente, all'aumentare del Recall (ovvero, rilevando più oggetti), la Precisione tende a diminuire. Questo grafico aiuta a visualizzare quanto bene il modello classifica i suoi rilevamenti in base alla copertura degli oggetti reali.
- **Curva di Recall (R Curve):** Questo grafico ('BoxR_curve') mostra l'andamento del Recall in funzione della soglia di confidenza (o, in alcuni contesti, del numero di falsi positivi ammessi). Idealmente, si desidera che il Recall rimanga elevato anche a

soglie di confidenza più alte, indicando che il modello riesce a catturare la maggior parte degli oggetti reali in modo affidabile.

- **Curva F1-Score (F1 Curve):** Questo grafico ('BoxF1_curve') rappresenta l'F1-score (media armonica di Precisione e Recall) in funzione della soglia di confidenza. L'F1-score è una metrica utile quando si desidera un bilanciamento tra Precisione e Recall. Il punto più alto di questa curva indica la soglia di confidenza ottimale per ottenere il miglior compromesso tra i due valori.

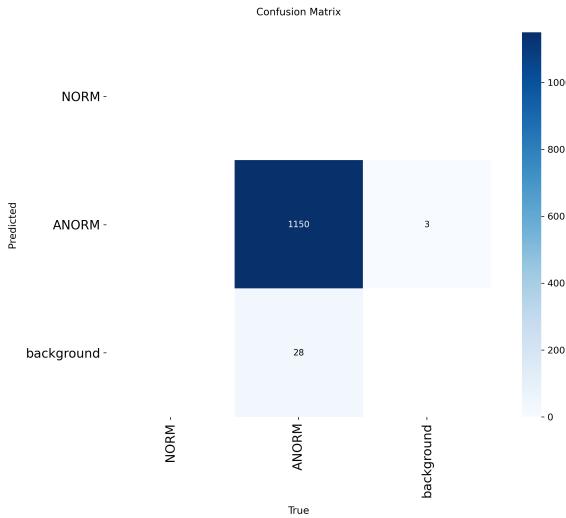


Figure 27: Matrice di Confusione del Modello

Da come abbiamo potuto vedere anche dalla confusion matrix 27, il nostro modello ha ottenuto risultati molto ottimi, evidenziando che la pipeline proposta fino a ora è risultata vincente.

5.4.3 Valutazione sul Test Set

Al termine dell'addestramento e della selezione del modello con i migliori iperparametri, è stata eseguita una valutazione finale e indipendente sul set di test. Questa fase è cruciale per stimare le prestazioni generalizzabili del modello su dati non visti. La valutazione è stata condotta utilizzando il comando specifico del framework YOLO per il testing, che elabora il set di test completo definito nel file 'data.yaml'.

La seguente tabella riassume le principali metriche di performance del modello YOLOv8n-p2 sul set di test. 28

5.5 Valutazione dell'Efficienza del Modello e Pipeline di Inferenza

Oltre alle metriche di accuratezza, è fondamentale valutare l'efficienza del modello in termini di tempo di elaborazione, specialmente per applicazioni in contesti clinici dove la rapidità di risposta può essere cruciale. La pipeline di inferenza del modello YOLOv8n-p2 è stata misurata per quantificare il tempo impiegato in ciascuna delle sue fasi principali, dall'acquisizione dell'immagine alla presentazione del risultato finale.

Table 10: Metriche di Valutazione del Modello YOLOv8n-p2 sul Test Set

Metrica	Valore (all classi)	Valore (ANORM)
Precisione (P)	0.975	0.975
Recall (R)	0.989	0.989
mAP50	0.989	0.989
mAP50-95	0.905	0.905
IoU medio	0.877	
FPPI	0.008	

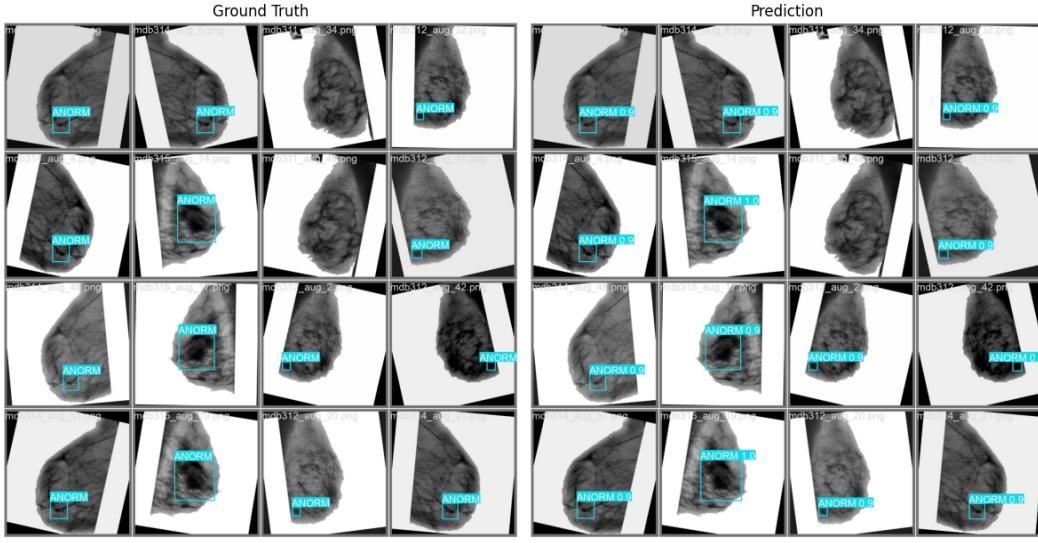


Figure 28: Esiti del modello con le relative confidenze

Il processo di inferenza per una singola immagine segue una pipeline ben definita, suddivisibile in diverse fasi che contribuiscono al risultato finale:

1. **Preprocessing Specifica per YOLO:** La prima fase coinvolge la preparazione dell’immagine di input per il modello YOLO. Questa preparazione è cruciale per garantire che l’immagine sia nel formato e nelle caratteristiche ottimali per essere elaborata dal modello. I passaggi includono:
 - **Caricamento e Ridimensionamento:** L’immagine mammografica originale viene caricata e ridimensionata a una dimensione standard, tipicamente 640x640 pixel, che è la dimensione di input attesa dal modello YOLOv8n-p2. Questo standardizza l’input per il modello.
 - **Filtro Gaussiano:** Viene applicato un filtro di sfocatura Gaussiano (con un kernel 5x5 nel caso specifico) per ridurre il rumore presente nell’immagine e ammorbidire le transizioni tra i pixel. Questo aiuta il modello a concentrarsi sulle caratteristiche salienti piuttosto che su dettagli irrilevanti o artefatti, rendendolo più robusto.
 - **Inversione dei Colori:** I valori dei pixel dell’immagine vengono invertiti (ad esempio, sottraendo il valore del pixel dal valore massimo possibile, come 255 per immagini a 8 bit). Nelle mammografie, l’inversione dei colori può migliorare

significativamente il contrasto delle lesioni, che spesso appaiono come aree più scure su uno sfondo più chiaro, rendendole più distinctive per il modello.

Il tempo impiegato per queste operazioni viene registrato accuratamente.

2. **Caricamento del Modello YOLO:** Prima di poter eseguire l'inferenza, il modello YOLOv8n-p2, pre-addestrato e ottimizzato, deve essere caricato in memoria. Questa fase include il caricamento dell'architettura del modello e dei suoi pesi. Il tempo di caricamento è un fattore importante per applicazioni che richiedono avvii rapidi o inferenze singole.
3. **Inferenza YOLO:** Questa è la fase centrale in cui il modello YOLO elabora l'immagine pre-processata e genera le previsioni dei bounding box e delle classi. Il modello analizza l'immagine per identificare le potenziali anomalie, assegnando una probabilità di confidenza a ciascun rilevamento. Il tempo impiegato per questa operazione è indicativo della velocità di elaborazione del modello.
4. **Preprocessing Specifica della Mammografia e Aggiustamento dei Bounding Box:** Dopo l'inferenza iniziale di YOLO, l'immagine originale (o una sua copia) subisce ulteriori passaggi di elaborazione volti a migliorare la sua qualità visiva e a facilitare l'interpretazione dei risultati in un contesto diagnostico:
 - **Rimozione dei Bordi Neri:** Le aree nere non informative attorno al tessuto mammario vengono rimosse per focalizzare l'attenzione sull'area di interesse diagnostico.
 - **Estrazione della Regione di Interesse (ROI):** La regione contenente il tessuto mammario principale viene isolata, eliminando sfondi e artefatti esterni. Questo assicura che l'analisi si concentri solo sulla parte rilevante della mammografia.
 - **Miglioramento del Contrasto:** Tecniche come il filtro mediano per la riduzione del rumore e l'Adaptive Histogram Equalization (CLAHE) vengono applicate per migliorare il contrasto e la visibilità delle strutture interne del seno, rendendo più evidenti le anomalie.
 - **Rimozione del Muscolo Pettorale:** Se presente, il muscolo pettorale viene identificato e rimosso o mascherato, in quanto può ostacolare il rilevamento delle anomalie e generare falsi positivi.

La misurazione del tempo di esecuzione per ciascuna di queste fasi consente di identificare i colli di bottiglia della pipeline e di valutare l'efficienza complessiva del sistema in un'applicazione pratica.

5.5.1 Tempi di Esecuzione della Pipeline di Inferenza

La valutazione dell'efficienza è stata condotta misurando il tempo impiegato da ciascuna fase della pipeline per un'immagine mammografica. I tempi registrati sono i seguenti:

- **Tempo totale:** Il tempo complessivo impiegato per processare una singola immagine, dalla sua lettura al salvataggio del risultato finale, è stato di **0.9650 secondi**.

- **Dettaglio per fase:**

- Preprocessing per YOLO: **0.0258s** (2.7% del tempo totale)
- Caricamento del modello: **0.3434s** (35.6% del tempo totale)
- Inferenza YOLO: **0.5959s** (61.7% del tempo totale)

In base a questi dati, si osserva che la fase di inferenza è la componente più dispendiosa in termini di tempo, seguita dal caricamento del modello. Il preprocessing iniziale per YOLO è estremamente rapido. Questo suggerisce che per applicazioni che richiedono un'elevata throughput o un'inferenza in tempo reale su più immagini consecutive (senza ricaricare il modello per ogni immagine), il sistema sarà molto efficiente una volta che il modello è già caricato in memoria, con l'inferenza che rappresenta il fattore dominante per il tempo di elaborazione per singola immagine.